My Project

Generated by Doxygen 1.10.0

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Namespace Documentation

4.1 buildteloc_ Namespace Reference

Classes

- struct t_buildtelocstruct

  generic parameter for a Teloc board
- struct t_teloc_config

4.2 compare_ Namespace Reference

4.3 config_ Namespace Reference

the namespace by Configuration file

Classes

- struct t_configstruct
- struct t_telocstrcut

4.3.1 Detailed Description

the namespace by Configuration file

4.4 configimpl_ Namespace Reference

the namespace used from the pimpl

Classes

- struct t_configstructimpl
- struct t_filestruct

  the scructure is the file row to fullfill

Functions

- t_configstructimpl * getaccescfgimpl (void)
- t_filestruct * getfilestruct (void)

4.4.1 Detailed Description

the namespace used from the pimpl

4.4.2 Function Documentation

4.4.2.1 getaccescfgimpl()

```
t_configstructimpl * configimpl_::getaccescfgimpl (
            void  )  [extern]
```

4.4.2.2 getfilestruct()

```
t_filestruct * configimpl_::getfilestruct (
            void  )  [extern]
```

4.5 debug_ Namespace Reference

namespace used to manages the debug

Functions

- void dbg_info (void)
- template<typename T >
  T dbg_display (T &x)
- void dummy (void)

Variables

- type_::UINT8 enable = 1

  variable used to enable the function to display the info

4.5.1 Detailed Description

namespace used to manages the debug

4.5.2 Function Documentation

4.5.2.1 dbg_display()

```
template<typename T >
T debug_::dbg_display (
             T & x )
```

Definition at line 40 of file Debug.h.

4.5.2.2 dbg_info()

```
void debug_::dbg_info (
             void  )
```

4.5.2.3 dummy()

```
void debug_::dummy (
             void  )
```

Definition at line 14 of file Debug.cpp.

4.5.3 Variable Documentation

4.5.3.1 enable

```
debug_::enable = 1  [extern]
```

variable used to enable the function to display the info

Definition at line 11 of file Debug.cpp.

4.6 teloc3000impl_ Namespace Reference

namspaxe teloc3000impl_

Enumerations

- enum eT3Code {
  NO_BOARD_ENABLED = 0 , DATRA_ENABLED = 0x04 , CAN_ENABLED = 0x08 , USCOA_ENABLED = 0x10 ,
  DAIOD_ENABLED = 0x20 , REBO_ENABLED = 0x40 , TACHA_ENABLED = 0x80 , USCOA_TACHA_ENABLED = 0x90 ,
  SABO_ENABLED = 0x100 , MVB_ENABLED = 0x200 , CPM_ENABLED = 0x400 }

  the structure manage the boards for the Teloc 3000

Functions

- eT3Code setSABO (type_::UINT64 maincode)

  the function set the SABO, if it's set on the old Teloc
- eT3Code setDIGITAL (type_::UINT64 maincode)

  the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc
- eT3Code setTACA (type_::UINT64 maincode)

  the function set the setTACA and the USCOA, if they are set on the old Teloc
- eT3Code setBUS (type_::UINT64 maincode)

  the function set the MVB or CAN bus , if they are set on the old Teloc
- eT3Code setCPM (type_::UINT64 maincode)

  the function set the CPM , if it's set on the old Teloc

4.6.1 Detailed Description

namspaxe teloc3000impl_

4.6.2 Enumeration Type Documentation

4.6.2.1 eT3Code

```
enum teloc3000impl_::eT3Code
```

the structure manage the boards for the Teloc 3000

Enumerator

NO_BOARD_ENABLED	NO_BOARD_ENABLED
DATRA_ENABLED	DATRA_ENABLED
CAN_ENABLED	CAN_ENABLED
USCOA_ENABLED	USCOA_ENABLED
DAIOD_ENABLED	DAIOD_ENABLED
REBO_ENABLED	REBO_ENABLED
TACHA_ENABLED	TACHA_ENABLED
USCOA_TACHA_ENABLED	USCOA_TACHA_ENABLED
SABO_ENABLED	SABO_ENABLED
MVB_ENABLED	SABO_ENABLED
CPM_ENABLED	CPM_ENABLED

Definition at line 21 of file Teloc3000_Impl.h.

4.6.3 Function Documentation

4.6.3.1 setBUS()

```
teloc3000impl_::eT3Code teloc3000impl_::setBUS (
            type_::UINT64 maincode )
```

the function set the MVB or CAN bus , if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 78 of file Teloc3000_Impl.cpp.

4.6.3.2 setCPM()

```
teloc3000impl_::eT3Code teloc3000impl_::setCPM (
            type_::UINT64 maincode )
```

the function set the CPM , if it's set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 100 of file Teloc3000_Impl.cpp.

4.6.3.3 setDIGITAL()

```
teloc3000impl_::eT3Code teloc3000impl_::setDIGITAL (
            type_::UINT64 maincode )
```

the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 56 of file Teloc3000_Impl.cpp.

4.6.3.4 setSABO()

```
teloc3000impl_::eT3Code teloc3000impl_::setSABO (
            type_::UINT64 maincode )
```

the function set the SABO, if it's set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 19 of file Teloc3000_Impl.cpp.

4.6.3.5 setTACA()

```
teloc3000impl_::eT3Code teloc3000impl_::setTACA (
            type_::UINT64 maincode )
```

the function set the setTACA and the USCOA, if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 33 of file Teloc3000_Impl.cpp.

4.7 teloc4000impl_ Namespace Reference

namespace teloc3000impl_

Enumerations

- enum eT4Code {
  NO_BOARD_ENABLED = 0 , ONLY_DRSCA_ENABLED = 0x08 , TECA_DRSCA_ENABLED = 0x0C , DAIOD_ENABLED = 0x10 ,
  DOCAA_ENABLED = 0x20 , SABOC_ENABLED = 0x40 , MVB_ENABLED = 0x80 , CAN_ENABLED = 0x100 ,
  GPS_ENABLED = 0x200 , CPM_ENABLED = 0x400 }

  the structure manage the board for the TEloc 4000

Functions

- eT4Code setDIGITAL (type_::UINT64 maincode)

  the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc
- eT4Code setSABO (type_::UINT64 maincode)

  the function set the SABOC, if it's set on the old Teloc
- eT4Code setTECA (type_::UINT64 maincode)

  the function set the setTECA and the DRSCA, if they are set on the old Teloc
- eT4Code setBUS (type_::UINT64 maincode)

  the function set the MVB or CAN bus , if they are set on the old Teloc
- eT4Code setGPS (type_::UINT64 maincode)

  the function set the GPS , if it's set on the old Teloc
- eT4Code setCPM (type_::UINT64 maincode)

  the function set the CPM , if it's set on the old Teloc

4.7.1 Detailed Description

namespace teloc3000impl_

4.7.2 Enumeration Type Documentation

4.7.2.1 eT4Code

```
enum teloc4000impl_::eT4Code
```

the structure manage the board for the TEloc 4000

Enumerator

NO_BOARD_ENABLED	NO_BOARD_ENABLED
ONLY_DRSCA_ENABLED	ONLY_DRSCA_ENABLED
TECA_DRSCA_ENABLED	TECA_DRSCA_ENABLED
DAIOD_ENABLED	DAIOD_ENABLED
DOCAA_ENABLED	DOCAA_ENABLED
SABOC_ENABLED	SABOC_ENABLED
MVB_ENABLED	MVB_ENABLED
CAN_ENABLED	CAN_ENABLED
GPS_ENABLED	GPS_ENABLED
CPM_ENABLED	CPM_ENABLED

Definition at line 17 of file Teloc4000_Impl.h.

4.7.3 Function Documentation

4.7.3.1 setBUS()

```
teloc4000impl_::eT4Code teloc4000impl_::setBUS (
            type_::UINT64 maincode )
```

the function set the MVB or CAN bus , if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 57 of file Teloc4000_Impl.cpp.

4.7.3.2 setCPM()

```
teloc4000impl_::eT4Code teloc4000impl_::setCPM (
            type_::UINT64 maincode )
```

the function set the CPM , if it's set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 87 of file Teloc4000_Impl.cpp.

4.7.3.3 setDIGITAL()

```
teloc4000impl_::eT4Code teloc4000impl_::setDIGITAL (
            type_::UINT64 maincode )
```

the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 35 of file Teloc4000_Impl.cpp.

4.7.3.4 setGPS()

```
teloc4000impl_::eT4Code teloc4000impl_::setGPS (
            type_::UINT64 maincode )
```

the function set the GPS , if it's set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 79 of file Teloc4000_Impl.cpp.

4.7.3.5 setSABO()

```
teloc4000impl_::eT4Code teloc4000impl_::setSABO (
            type_::UINT64 maincode )
```

the function set the SABOC, if it's set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 11 of file Teloc4000_Impl.cpp.

4.7.3.6 setTECA()

```
teloc4000impl_::eT4Code teloc4000impl_::setTECA (
            type_::UINT64 maincode )
```

the function set the setTECA and the DRSCA, if they are set on the old Teloc

Parameters

maincode,code	of assembly code under check

Returns

TRUE or FALSE

Definition at line 18 of file Teloc4000_Impl.cpp.

4.8 type_ Namespace Reference

namespace used to manages the typedef

Typedefs

- typedef unsigned short UINT16
- typedef unsigned int UINT64
- typedef unsigned char UINT8
- typedef char CHAR

Enumerations

- enum ebool { FALSE , TRUE }
- enum e_result { RESULT_OK , RESULT_POINTER_NOT_ADDRESSED , RESULT_OUT_OF_RANGE , RESULT_NOT_READY_UART = 4 }

  the enum is used as validity check in the methods

4.8.1 Detailed Description

namespace used to manages the typedef

4.8.2 Typedef Documentation

4.8.2.1 CHAR

```
typedef char type_::CHAR
```

Definition at line 23 of file Types.h.

4.8.2.2 UINT16

```
typedef unsigned short type_::UINT16
```

Definition at line 20 of file Types.h.

4.8.2.3 UINT64

```
typedef unsigned int type_::UINT64
```

Definition at line 21 of file Types.h.

4.8.2.4 UINT8

```
typedef unsigned char type_::UINT8
```

Definition at line 22 of file Types.h.

4.8.3 Enumeration Type Documentation

4.8.3.1 e_result

```
enum type_::e_result
```

the enum is used as validity check in the methods

Enumerator

RESULT_OK	Result of check ok
RESULT_POINTER_NOT_ADDRESSED	Result of pointer check ko
RESULT_OUT_OF_RANGE	Result of range check ko
RESULT_NOT_READY_UART	Result of uart not ready

Definition at line 38 of file Types.h.

4.8.3.2 ebool

```
enum type_::ebool
```

Enumerator

FALSE	False is 0
TRUE	True is 1

Definition at line 28 of file Types.h.

4.9 util_ Namespace Reference

Functions

- type_::ebool charpointer_compare (const type_::CHAR ∗a, const type_::CHAR ∗b)
- type_::e_result CheckArg (void ∗pArg)

  the function checks if the pointer is addressed
- type_::UINT64 ConverTelocCode2Num (std::string teloccode)

  the function returns which kinf of Teloc is under test

4.9.1 Function Documentation

4.9.1.1 charpointer_compare()

```
type_::ebool util_::charpointer_compare (
            const type_::CHAR * a,
            const type_::CHAR * b )
```

Definition at line 11 of file Util.cpp.

4.9.1.2 CheckArg()

```
type_::e_result util_::CheckArg (
            void * pArg )
```

the function checks if the pointer is addressed

Parameters

∗pArg,pointer	to be checked

Returns

status of check

Definition at line 22 of file Util.cpp.

4.9.1.3 ConverTelocCode2Num()

```
type_::UINT64 util_::ConverTelocCode2Num (
            std::string teloccode )
```

the function returns which kinf of Teloc is under test

Parameters

∗teloccode,teloc	code

Returns

kind of Teloc

Definition at line 27 of file Util.cpp.

Chapter 5

Class Documentation

5.1 CodeT4< T > Class Template Reference

Private Attributes

- T T4Code

5.1.1 Detailed Description

template<class T>
class CodeT4< T >

Definition at line 17 of file ServiceTool.cpp.

5.1.2 Member Data Documentation

5.1.2.1 T4Code

```
template<class T >
T CodeT4< T >::T4Code  [private]
```

Definition at line 20 of file ServiceTool.cpp.

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

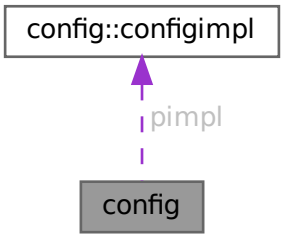
- ServiceTool.cpp

5.2 config Class Reference

the class manages the methods to parser the configuration file

`#include <Configuration.h>`

Collaboration diagram for config:



Classes

- struct configimpl

Public Member Functions

- void readfileconfig (void)

Static Public Member Functions

- static config & getinstance ()

Static Protected Member Functions

- static config_::t_configstruct * getconfigstruct (void)
- static config_::t_telocstrcut * gettelocstruct (void)

Private Member Functions

- config ()
- virtual ~config ()
- config (const config &)
- const config & operator= (const config &)
- std::string whoaml (std::string line)

Private Attributes

- configimpl * pimpl

5.2.1 Detailed Description

the class manages the methods to parser the configuration file

Definition at line 46 of file Configuration.h.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 config() [1/2]

```
config::config ( )  [private]
```

Definition at line 26 of file Configuration.cpp.

5.2.2.2 ~config()

```
virtual config::~config ( )  [inline], [private], [virtual]
```

Definition at line 55 of file Configuration.h.

5.2.2.3 config() [2/2]

```
config::config (
            const config &  )  [private]
```

5.2.3 Member Function Documentation

5.2.3.1 getconfigstruct()

```
config_::t_configstruct * config::getconfigstruct (
            void  )  [static], [protected]
```

Definition at line 34 of file Configuration.cpp.

5.2.3.2 getinstance()

```
config & config::getinstance ( )  [static]
```

Definition at line 46 of file Configuration.cpp.

5.2.3.3 gettelocstruct()

```
config_::t_telocstrcut * config::gettelocstruct (
            void  )  [static], [protected]
```

Definition at line 40 of file Configuration.cpp.

5.2.3.4 operator=()

```
const config & config::operator= (
            const config &  )  [private]
```

5.2.3.5 readfileconfig()

```
void config::readfileconfig (
            void  )
```

Definition at line 56 of file Configuration.cpp.

5.2.3.6 whoaml()

```
string config::whoamI (
            std::string line )  [private]
```

Definition at line 140 of file Configuration.cpp.

5.2.4 Member Data Documentation

5.2.4.1 pimpl

```
configimpl* config::pimpl  [private]
```

Definition at line 53 of file Configuration.h.

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

- include/Configuration.h
- Configuration.cpp

5.3 config::configimpl Struct Reference

```
#include <Configuration_impl.h>
```

Public Member Functions

- configimpl ()
- virtual ~configimpl ()
- void extract_filename (std::string line, type_::CHAR *filename, type_::CHAR *assemblycode)
- void extract_column_compare (std::string line, type_::CHAR *col)
- type_::ebool find_column (const type_::CHAR *title, const type_::CHAR *col)
- type_::ebool parser_kenfile (const type_::CHAR *col, std::string filename)
- type_::UINT64 getsizeTeloc (void)
- type_::UINT64 create_T4code (type_::UINT64 main_code)
- type_::UINT64 create_T3code (type_::UINT64 main_code)
- void scroll_column (const std::string teloccode)
- void create_template (ofstream &osheet, std::string teloc)
- void create_teloc_assembly (const char *s, Sheet *osheet, type_::UINT64 row)
- void create_output_file (std::vector< std::string > col, ofstream &file)
- void extract_family (std::string code, ofstream &osheet, configimpl_::t_filestruct *ptr)
- void extract_version (std::string code, std::string &variant)
- void write_variant (type_::UINT64 pos, std::string variant, configimpl_::t_filestruct *ptr)
- void compare_create_configuration (fstream &osheet)

5.3.1 Detailed Description

Definition at line 88 of file Configuration_impl.h.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 configimpl()

```
config::configimpl::configimpl ( )  [inline]
```

Definition at line 90 of file Configuration_impl.h.

5.3.2.2 ~configimpl()

```
virtual config::configimpl::~configimpl ( )  [inline], [virtual]
```

Definition at line 91 of file Configuration_impl.h.

5.3.3 Member Function Documentation

5.3.3.1 compare_create_configuration()

```
void config::configimpl::compare_create_configuration (
            fstream & osheet )
```

Definition at line 537 of file Configuration_impl.cpp.

5.3.3.2 create_output_file()

```
void config::configimpl::create_output_file (
            std::vector< std::string > col,
            ofstream & file )
```

Definition at line 314 of file Configuration_impl.cpp.

5.3.3.3 create_T3code()

```
type_::UINT64 config::configimpl::create_T3code (
            type_::UINT64 main_code )
```

Definition at line 518 of file Configuration_impl.cpp.

5.3.3.4 create_T4code()

```
type_::UINT64 config::configimpl::create_T4code (
            type_::UINT64 main_code )
```

Definition at line 502 of file Configuration_impl.cpp.

5.3.3.5 create_teloc_assembly()

```
void config::configimpl::create_teloc_assembly (
            const char * s,
            Sheet * osheet,
            type_::UINT64 row )
```

Definition at line 241 of file Configuration_impl.cpp.

5.3.3.6 create_template()

```
void config::configimpl::create_template (
            ofstream & osheet,
            std::string teloc )
```

Definition at line 224 of file Configuration_impl.cpp.

5.3.3.7 extract_column_compare()

```
void config::configimpl::extract_column_compare (
            std::string line,
            type_::CHAR * col )
```

Definition at line 77 of file Configuration_impl.cpp.

5.3.3.8 extract_family()

```
void config::configimpl::extract_family (
            std::string code,
            ofstream & osheet,
            configimpl_::t_filestruct * ptr )
```

Definition at line 355 of file Configuration_impl.cpp.

5.3.3.9 extract_filename()

```
void config::configimpl::extract_filename (
            std::string line,
            type_::CHAR * filename,
            type_::CHAR * assemblycode )
```

Definition at line 58 of file Configuration_impl.cpp.

5.3.3.10 extract_version()

```
void config::configimpl::extract_version (
            std::string code,
            std::string & variant )
```

Definition at line 346 of file Configuration_impl.cpp.

5.3.3.11 find_column()

```
type_::ebool config::configimpl::find_column (
            const type_::CHAR * title,
            const type_::CHAR * col )
```

Definition at line 91 of file Configuration_impl.cpp.

5.3.3.12 getsizeTeloc()

```
type_::UINT64 config::configimpl::getsizeTeloc (
            void  )
```

Definition at line 167 of file Configuration_impl.cpp.

5.3.3.13 parser_kenfile()

```
type_::ebool config::configimpl::parser_kenfile (
            const type_::CHAR * col,
            std::string filename )
```

Definition at line 104 of file Configuration_impl.cpp.

5.3.3.14 scroll_column()

```
void config::configimpl::scroll_column (
            const std::string teloccode )
```

!!

Definition at line 140 of file Configuration_impl.cpp.

5.3.3.15 write_variant()

```
void config::configimpl::write_variant (
            type_::UINT64 pos,
            std::string variant,
            configimpl_::t_filestruct * ptr )
```

Definition at line 384 of file Configuration_impl.cpp.

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

- include/Configuration_impl.h
- Configuration_impl.cpp

5.4 configimpl Class Reference

the pimpl class manages the methods hide in the configuration class

`#include <Configuration_impl.h>`

5.4.1 Detailed Description

the pimpl class manages the methods hide in the configuration class

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

- include/Configuration_impl.h

5.5 debug Class Reference

`#include <Debug.h>`

Public Member Functions

- debug ()
- void debuginfo (void)
- virtual ~debug ()

5.5.1 Detailed Description

Definition at line 47 of file Debug.h.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 debug()

```
debug::debug ( )
```

5.5.2.2 ~debug()

```
virtual debug::~debug ( )  [inline], [virtual]
```

Definition at line 52 of file Debug.h.

5.5.3 Member Function Documentation

5.5.3.1 debuginfo()

```
void debug::debuginfo (
             void  )
```

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

- include/Debug.h

5.6 buildteloc_::t_buildtelocstruct Struct Reference

generic parameter for a Teloc board

```
#include <BuildTeloc.h>
```

Public Attributes

- std::string board_name
- type_::ebool active
- type_::UINT64 numberofboard

5.6.1 Detailed Description

generic parameter for a Teloc board

Definition at line 157 of file BuildTeloc.h.

5.6.2 Member Data Documentation

5.6.2.1 active

`type_::ebool buildteloc_::t_buildtelocstruct::active`

present or not

Definition at line 160 of file BuildTeloc.h.

5.6.2.2 board_name

`std::string buildteloc_::t_buildtelocstruct::board_name`

name of board

Definition at line 159 of file BuildTeloc.h.

5.6.2.3 numberofboard

`type_::UINT64 buildteloc_::t_buildtelocstruct::numberofboard`

how many board

Definition at line 161 of file BuildTeloc.h.

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

- include/BuildTeloc.h

5.7 config_::t_configstruct Struct Reference

`#include <Configuration.h>`

Public Attributes

- type_::UINT64 index_row
- std::string line
- std::string title
- type_::CHAR filename [20]
- type_::CHAR column [60]
- type_::UINT64 numberboardTeloc [20]
- type_::CHAR assemblycode [20][20]

5.7.1 Detailed Description

Definition at line 19 of file Configuration.h.

5.7.2 Member Data Documentation

5.7.2.1 assemblycode

```
type_::CHAR config_::t_configstruct::assemblycode[20][20]
```

assembly code

Definition at line 27 of file Configuration.h.

5.7.2.2 column

```
type_::CHAR config_::t_configstruct::column[60]
```

Definition at line 25 of file Configuration.h.

5.7.2.3 filename

```
type_::CHAR config_::t_configstruct::filename[20]
```

Definition at line 24 of file Configuration.h.

5.7.2.4 index_row

```
type_::UINT64 config_::t_configstruct::index_row
```

Definition at line 21 of file Configuration.h.

5.7.2.5 line

```
std::string config_::t_configstruct::line
```

Definition at line 22 of file Configuration.h.

5.7.2.6 numberboardTeloc

```
type_::UINT64 config_::t_configstruct::numberboardTeloc[20]
```

number of board of Teloc

Definition at line 26 of file Configuration.h.

5.7.2.7 title

```
std::string config_::t_configstruct::title
```

Definition at line 23 of file Configuration.h.

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

- include/Configuration.h

5.8 configimpl_::t_configstructimpl Struct Reference

```
#include <Configuration_impl.h>
```

Public Attributes

- type_::UINT64 findcolumn

5.8.1 Detailed Description

Definition at line 49 of file Configuration_impl.h.

5.8.2 Member Data Documentation

5.8.2.1 findcolumn

```
type_::UINT64 configimpl_::t_configstructimpl::findcolumn
```

column to scroll in exel file

Definition at line 51 of file Configuration_impl.h.

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

- include/Configuration_impl.h

5.9 configimpl_::t_filestruct Struct Reference

the scructure is the file row to fullfill

```
#include <Configuration_impl.h>
```

Public Attributes

- std::string assembly_code
- std::string customer
- std::string posu
- std::string core
- std::string ioco
- std::string daio
- std::string rebo
- std::string sabo
- std::string mvb
- std::string can
- std::string gps
- std::string cpm
- std::string sram
- std::string flash
- std::string backplane
- std::string datra

5.9.1 Detailed Description

the scructure is the file row to fullfill

Definition at line 58 of file Configuration_impl.h.

5.9.2 Member Data Documentation

5.9.2.1 assembly_code

```
std::string configimpl_::t_filestruct::assembly_code
```

assembly code

Definition at line 60 of file Configuration_impl.h.

5.9.2.2 backplane

```
std::string configimpl_::t_filestruct::backplane
```

backplane code

Definition at line 74 of file Configuration_impl.h.

5.9.2.3 can

```
std::string configimpl_::t_filestruct::can
```

can code

Definition at line 69 of file Configuration_impl.h.

5.9.2.4 core

```
std::string configimpl_::t_filestruct::core
```

core code

Definition at line 63 of file Configuration_impl.h.

5.9.2.5 cpm

```
std::string configimpl_::t_filestruct::cpm
```

cpm code

Definition at line 71 of file Configuration_impl.h.

5.9.2.6 customer

```
std::string configimpl_::t_filestruct::customer
```

customer

Definition at line 61 of file Configuration_impl.h.

5.9.2.7 daio

```
std::string configimpl_::t_filestruct::daio
```

daio code

Definition at line 65 of file Configuration_impl.h.

5.9.2.8 datra

```
std::string configimpl_::t_filestruct::datra
```

datra code

Definition at line 75 of file Configuration_impl.h.

5.9.2.9 flash

```
std::string configimpl_::t_filestruct::flash
```

flash code

Definition at line 73 of file Configuration_impl.h.

5.9.2.10 gps

```
std::string configimpl_::t_filestruct::gps
```

gps code

Definition at line 70 of file Configuration_impl.h.

5.9.2.11 ioco

```
std::string configimpl_::t_filestruct::ioco
```

ioco code

Definition at line 64 of file Configuration_impl.h.

5.9.2.12 mvb

```
std::string configimpl_::t_filestruct::mvb
```

mvb code

Definition at line 68 of file Configuration_impl.h.

5.9.2.13 posu

```
std::string configimpl_::t_filestruct::posu
```

posu code

Definition at line 62 of file Configuration_impl.h.

5.9.2.14 rebo

```
std::string configimpl_::t_filestruct::rebo
```

rebo code

Definition at line 66 of file Configuration_impl.h.

5.9.2.15 sabo

```
std::string configimpl_::t_filestruct::sabo
```

sabo code

Definition at line 67 of file Configuration_impl.h.

5.9.2.16 sram

```
std::string configimpl_::t_filestruct::sram
```

sram code

Definition at line 72 of file Configuration_impl.h.

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

- include/Configuration_impl.h

5.10 buildteloc_::t_teloc_config Struct Reference

```
#include <BuildTeloc.h>
```

Public Attributes

- type_::UINT64 matchvalue
- type_::UINT64 posu
- type_::UINT64 core
- type_::UINT64 ioco
- type_::UINT64 daio
- type_::UINT64 rebo
- type_::UINT64 sabo
- type_::UINT64 mvb
- type_::UINT64 can
- type_::UINT64 gps
- type_::UINT64 cpm
- type_::UINT64 sram
- type_::UINT64 flash
- type_::UINT64 backplane

5.10.1 Detailed Description

Definition at line 163 of file BuildTeloc.h.

5.10.2 Member Data Documentation

5.10.2.1 backplane

```
type_::UINT64 buildteloc_::t_teloc_config::backplane
```

value match backplane

Definition at line 178 of file BuildTeloc.h.

5.10.2.2 can

```
type_::UINT64 buildteloc_::t_teloc_config::can
```

value match mvb

Definition at line 173 of file BuildTeloc.h.

5.10.2.3 core

```
type_::UINT64 buildteloc_::t_teloc_config::core
```

value match core

Definition at line 167 of file BuildTeloc.h.

5.10.2.4 cpm

```
type_::UINT64 buildteloc_::t_teloc_config::cpm
```

value match cpm

Definition at line 175 of file BuildTeloc.h.

5.10.2.5 daio

```
type_::UINT64 buildteloc_::t_teloc_config::daio
```

value match daio

Definition at line 169 of file BuildTeloc.h.

5.10.2.6 flash

```
type_::UINT64 buildteloc_::t_teloc_config::flash
```

value match flash

Definition at line 177 of file BuildTeloc.h.

5.10.2.7 gps

```
type_::UINT64 buildteloc_::t_teloc_config::gps
```

value match mvb

Definition at line 174 of file BuildTeloc.h.

5.10.2.8 ioco

`type_::UINT64 buildteloc_::t_teloc_config::ioco`

value match ioco

Definition at line 168 of file BuildTeloc.h.

5.10.2.9 matchvalue

`type_::UINT64 buildteloc_::t_teloc_config::matchvalue`

value match

Definition at line 165 of file BuildTeloc.h.

5.10.2.10 mvb

`type_::UINT64 buildteloc_::t_teloc_config::mvb`

value match mvb

Definition at line 172 of file BuildTeloc.h.

5.10.2.11 posu

`type_::UINT64 buildteloc_::t_teloc_config::posu`

value match posu

Definition at line 166 of file BuildTeloc.h.

5.10.2.12 rebo

`type_::UINT64 buildteloc_::t_teloc_config::rebo`

value match rebo

Definition at line 170 of file BuildTeloc.h.

5.10.2.13 sabo

`type_::UINT64 buildteloc_::t_teloc_config::sabo`

value match sabo

Definition at line 171 of file BuildTeloc.h.

5.10.2.14 sram

```
type_::UINT64 buildteloc_::t_teloc_config::sram
```

value match sram

Definition at line 176 of file BuildTeloc.h.

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

- include/BuildTeloc.h

5.11 config_::t_telocstrcut Struct Reference

```
#include <Configuration.h>
```

Public Attributes

- std::string kindofTeloc
- type_::UINT64 Teloc

5.11.1 Detailed Description

Definition at line 35 of file Configuration.h.

5.11.2 Member Data Documentation

5.11.2.1 kindofTeloc

```
std::string config_::t_telocstrcut::kindofTeloc
```

kinf of teloc

Definition at line 37 of file Configuration.h.

5.11.2.2 Teloc

```
type_::UINT64 config_::t_telocstrcut::Teloc
```

0->Teloc1500; 1->Teloc2500

Definition at line 38 of file Configuration.h.

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

- include/Configuration.h

Chapter 6

File Documentation

6.1 build/cmake.debug.linux.x86_64/CMakeFiles/3.25.1/CompilerIdC/↩ CMakeCCompilerId.c File Reference

Macros

- #define __has_include(x) 0
- #define COMPILER_ID ""
- #define STRINGIFY_HELPER(X) #X
- #define STRINGIFY(X) STRINGIFY_HELPER(X)
- #define PLATFORM_ID
- #define ARCHITECTURE_ID
- #define DEC(n)
- #define HEX(n)
- #define C_VERSION

Functions

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

Variables

- char const * info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
- char const * info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
- char const * info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
- const char * info_language_standard_default
- const char * info_language_extensions_default

6.1.1 Macro Definition Documentation

6.1.1.1 __has_include

```
#define __has_include(
              x ) 0
```

Definition at line 17 of file CMakeCCompilerId.c.

6.1.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

Definition at line 718 of file CMakeCCompilerId.c.

6.1.1.3 C_VERSION

```
#define C_VERSION
```

Definition at line 807 of file CMakeCCompilerId.c.

6.1.1.4 COMPILER_ID

```
#define COMPILER_ID ""
```

Definition at line 429 of file CMakeCCompilerId.c.

6.1.1.5 DEC

```
#define DEC(
              n )
```

Value:

```
  ('0' + (((n) / 10000000)%10)), \
  ('0' + (((n) / 1000000)%10)),  \
  ('0' + (((n) / 100000)%10)),   \
  ('0' + (((n) / 10000)%10)),    \
  ('0' + (((n) / 1000)%10)),     \
  ('0' + (((n) / 100)%10)),      \
  ('0' + (((n) / 10)%10)),       \
  ('0' +  ((n) % 10))
```

Definition at line 722 of file CMakeCCompilerId.c.

6.1.1.6 HEX

```
#define HEX(
              n )
```

Value:

```
  ('0' + ((n)>>28 & 0xF)), \
  ('0' + ((n)>>24 & 0xF)), \
  ('0' + ((n)>>20 & 0xF)), \
  ('0' + ((n)>>16 & 0xF)), \
  ('0' + ((n)>>12 & 0xF)), \
  ('0' + ((n)>>8  & 0xF)), \
  ('0' + ((n)>>4  & 0xF)), \
  ('0' + ((n)     & 0xF))
```

Definition at line 733 of file CMakeCCompilerId.c.

6.1.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```

Definition at line 560 of file CMakeCCompilerId.c.

6.1.1.8 STRINGIFY

```
#define STRINGIFY(
              X ) STRINGIFY_HELPER(X)
```

Definition at line 450 of file CMakeCCompilerId.c.

6.1.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(
              X ) #X
```

Definition at line 449 of file CMakeCCompilerId.c.

6.1.2 Function Documentation

6.1.2.1 main()

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

Definition at line 841 of file CMakeCCompilerId.c.

6.1.3 Variable Documentation

6.1.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

Definition at line 799 of file CMakeCCompilerId.c.

6.1.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

Definition at line 436 of file CMakeCCompilerId.c.

6.1.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["



  "OFF"

"]"
```

Definition at line 823 of file CMakeCCompilerId.c.

6.1.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
=
  "INFO" ":" "standard_default[" C_VERSION "]"
```

Definition at line 820 of file CMakeCCompilerId.c.

6.1.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

Definition at line 798 of file CMakeCCompilerId.c.

6.2 CMakeCCompilerId.c

Go to the documentation of this file.

```
00001 #ifdef __cplusplus
00002 # error "A C++ compiler has been selected for C."
00003 #endif
00004
00005 #if defined(__18CXX)
00006 # define ID_VOID_MAIN
00007 #endif
00008 #if defined(__CLASSIC_C__)
00009 /* cv-qualifiers did not exist in K&R C */
00010 # define const
00011 # define volatile
00012 #endif
00013
00014 #if !defined(__has_include)
00015 /* If the compiler does not have __has_include, pretend the answer is
00016    always no.  */
00017 #  define __has_include(x) 0
00018 #endif
00019
00020
00021 /* Version number components: V=Version, R=Revision, P=Patch
00022    Version date components:   YYYY=Year, MM=Month,   DD=Day  */
00023
00024 #if defined(__INTEL_COMPILER) || defined(__ICC)
00025 # define COMPILER_ID "Intel"
00026 # if defined(_MSC_VER)
00027 #  define SIMULATE_ID "MSVC"
00028 # endif
00029 # if defined(__GNUC__)
```

```
00030 #  define SIMULATE_ID "GNU"
00031 # endif
00032   /* __INTEL_COMPILER = VRP prior to 2021, and then VVVV for 2021 and later,
00033      except that a few beta releases use the old format with V=2021.  */
00034 # if __INTEL_COMPILER < 2021 || __INTEL_COMPILER == 202110 || __INTEL_COMPILER == 202111
00035 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER/100)
00036 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER/10 % 10)
00037 #  if defined(__INTEL_COMPILER_UPDATE)
00038 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER_UPDATE)
00039 #  else
00040 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER   % 10)
00041 #  endif
00042 # else
00043 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER)
00044 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER_UPDATE)
00045    /* The third version component from --version is an update index,
00046       but no macro is provided for it.  */
00047 #  define COMPILER_VERSION_PATCH DEC(0)
00048 # endif
00049 # if defined(__INTEL_COMPILER_BUILD_DATE)
00050    /* __INTEL_COMPILER_BUILD_DATE = YYYYMMDD */
00051 #  define COMPILER_VERSION_TWEAK DEC(__INTEL_COMPILER_BUILD_DATE)
00052 # endif
00053 # if defined(_MSC_VER)
00054    /* _MSC_VER = VVRR */
00055 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00056 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00057 # endif
00058 # if defined(__GNUC__)
00059 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00060 # elif defined(__GNUG__)
00061 #  define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00062 # endif
00063 # if defined(__GNUC_MINOR__)
00064 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00065 # endif
00066 # if defined(__GNUC_PATCHLEVEL__)
00067 #  define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00068 # endif
00069
00070 #elif (defined(__clang__) && defined(__INTEL_CLANG_COMPILER)) || defined(__INTEL_LLVM_COMPILER)
00071 # define COMPILER_ID "IntelLLVM"
00072 #if defined(_MSC_VER)
00073 # define SIMULATE_ID "MSVC"
00074 #endif
00075 #if defined(__GNUC__)
00076 # define SIMULATE_ID "GNU"
00077 #endif
00078 /* __INTEL_LLVM_COMPILER = VVVVRP prior to 2021.2.0, VVVVRRPP for 2021.2.0 and
00079  * later.  Look for 6 digit vs. 8 digit version number to decide encoding.
00080  * VVVV is no smaller than the current year when a version is released.
00081  */
00082 #if __INTEL_LLVM_COMPILER < 1000000L
00083 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/100)
00084 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/10 % 10)
00085 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER    % 10)
00086 #else
00087 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/10000)
00088 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/100 % 100)
00089 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER     % 100)
00090 #endif
00091 #if defined(_MSC_VER)
00092   /* _MSC_VER = VVRR */
00093 # define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00094 # define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00095 #endif
00096 #if defined(__GNUC__)
00097 # define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00098 #elif defined(__GNUG__)
00099 # define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00100 #endif
00101 #if defined(__GNUC_MINOR__)
00102 # define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00103 #endif
00104 #if defined(__GNUC_PATCHLEVEL__)
00105 # define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00106 #endif
00107
00108 #elif defined(__PATHCC__)
00109 # define COMPILER_ID "PathScale"
00110 # define COMPILER_VERSION_MAJOR DEC(__PATHCC__)
00111 # define COMPILER_VERSION_MINOR DEC(__PATHCC_MINOR__)
00112 # if defined(__PATHCC_PATCHLEVEL__)
00113 #  define COMPILER_VERSION_PATCH DEC(__PATHCC_PATCHLEVEL__)
00114 # endif
00115
00116 #elif defined(__BORLANDC__) && defined(__CODEGEARC_VERSION__)
```

```
00117 # define COMPILER_ID "Embarcadero"
00118 # define COMPILER_VERSION_MAJOR HEX(__CODEGEARC_VERSION__»24 & 0x00FF)
00119 # define COMPILER_VERSION_MINOR HEX(__CODEGEARC_VERSION__»16 & 0x00FF)
00120 # define COMPILER_VERSION_PATCH DEC(__CODEGEARC_VERSION__     & 0xFFFF)
00121
00122 #elif defined(__BORLANDC__)
00123 # define COMPILER_ID "Borland"
00124   /* __BORLANDC__ = 0xVRR */
00125 # define COMPILER_VERSION_MAJOR HEX(__BORLANDC__»8)
00126 # define COMPILER_VERSION_MINOR HEX(__BORLANDC__ & 0xFF)
00127
00128 #elif defined(__WATCOMC__) && __WATCOMC__ < 1200
00129 # define COMPILER_ID "Watcom"
00130    /* __WATCOMC__ = VVRR */
00131 # define COMPILER_VERSION_MAJOR DEC(__WATCOMC__ / 100)
00132 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00133 # if (__WATCOMC__ % 10) > 0
00134 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00135 # endif
00136
00137 #elif defined(__WATCOMC__)
00138 # define COMPILER_ID "OpenWatcom"
00139    /* __WATCOMC__ = VVRP + 1100 */
00140 # define COMPILER_VERSION_MAJOR DEC((__WATCOMC__ - 1100) / 100)
00141 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00142 # if (__WATCOMC__ % 10) > 0
00143 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00144 # endif
00145
00146 #elif defined(__SUNPRO_C)
00147 # define COMPILER_ID "SunPro"
00148 # if __SUNPRO_C >= 0x5100
00149    /* __SUNPRO_C = 0xVRRP */
00150 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_C»12)
00151 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_C»4 & 0xFF)
00152 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_C    & 0xF)
00153 # else
00154    /* __SUNPRO_CC = 0xVRP */
00155 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_C»8)
00156 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_C»4 & 0xF)
00157 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_C    & 0xF)
00158 # endif
00159
00160 #elif defined(__HP_cc)
00161 # define COMPILER_ID "HP"
00162   /* __HP_cc = VVRRPP */
00163 # define COMPILER_VERSION_MAJOR DEC(__HP_cc/10000)
00164 # define COMPILER_VERSION_MINOR DEC(__HP_cc/100 % 100)
00165 # define COMPILER_VERSION_PATCH DEC(__HP_cc     % 100)
00166
00167 #elif defined(__DECC)
00168 # define COMPILER_ID "Compaq"
00169   /* __DECC_VER = VVRRTPPPP */
00170 # define COMPILER_VERSION_MAJOR DEC(__DECC_VER/10000000)
00171 # define COMPILER_VERSION_MINOR DEC(__DECC_VER/100000  % 100)
00172 # define COMPILER_VERSION_PATCH DEC(__DECC_VER         % 10000)
00173
00174 #elif defined(__IBMC__) && defined(__COMPILER_VER__)
00175 # define COMPILER_ID "zOS"
00176   /* __IBMC__ = VRP */
00177 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00178 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00179 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00180
00181 #elif defined(__open_xl__) && defined(__clang__)
00182 # define COMPILER_ID "IBMClang"
00183 # define COMPILER_VERSION_MAJOR DEC(__open_xl_version__)
00184 # define COMPILER_VERSION_MINOR DEC(__open_xl_release__)
00185 # define COMPILER_VERSION_PATCH DEC(__open_xl_modification__)
00186 # define COMPILER_VERSION_TWEAK DEC(__open_xl_ptf_fix_level__)
00187
00188
00189 #elif defined(__ibmxl__) && defined(__clang__)
00190 # define COMPILER_ID "XLClang"
00191 # define COMPILER_VERSION_MAJOR DEC(__ibmxl_version__)
00192 # define COMPILER_VERSION_MINOR DEC(__ibmxl_release__)
00193 # define COMPILER_VERSION_PATCH DEC(__ibmxl_modification__)
00194 # define COMPILER_VERSION_TWEAK DEC(__ibmxl_ptf_fix_level__)
00195
00196
00197 #elif defined(__IBMC__) && !defined(__COMPILER_VER__) && __IBMC__ >= 800
00198 # define COMPILER_ID "XL"
00199   /* __IBMC__ = VRP */
00200 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00201 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00202 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00203
```

```
00204 #elif defined(__IBMC__) && !defined(__COMPILER_VER__) && __IBMC__ < 800
00205 # define COMPILER_ID "VisualAge"
00206   /* __IBMC__ = VRP */
00207 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00208 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00209 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00210
00211 #elif defined(__NVCOMPILER)
00212 # define COMPILER_ID "NVHPC"
00213 # define COMPILER_VERSION_MAJOR DEC(__NVCOMPILER_MAJOR__)
00214 # define COMPILER_VERSION_MINOR DEC(__NVCOMPILER_MINOR__)
00215 # if defined(__NVCOMPILER_PATCHLEVEL__)
00216 #  define COMPILER_VERSION_PATCH DEC(__NVCOMPILER_PATCHLEVEL__)
00217 # endif
00218
00219 #elif defined(__PGI)
00220 # define COMPILER_ID "PGI"
00221 # define COMPILER_VERSION_MAJOR DEC(__PGIC__)
00222 # define COMPILER_VERSION_MINOR DEC(__PGIC_MINOR__)
00223 # if defined(__PGIC_PATCHLEVEL__)
00224 #  define COMPILER_VERSION_PATCH DEC(__PGIC_PATCHLEVEL__)
00225 # endif
00226
00227 #elif defined(_CRAYC)
00228 # define COMPILER_ID "Cray"
00229 # define COMPILER_VERSION_MAJOR DEC(_RELEASE_MAJOR)
00230 # define COMPILER_VERSION_MINOR DEC(_RELEASE_MINOR)
00231
00232 #elif defined(__TI_COMPILER_VERSION__)
00233 # define COMPILER_ID "TI"
00234   /* __TI_COMPILER_VERSION__ = VVVRRRPPP */
00235 # define COMPILER_VERSION_MAJOR DEC(__TI_COMPILER_VERSION__/1000000)
00236 # define COMPILER_VERSION_MINOR DEC(__TI_COMPILER_VERSION__/1000   % 1000)
00237 # define COMPILER_VERSION_PATCH DEC(__TI_COMPILER_VERSION__        % 1000)
00238
00239 #elif defined(__CLANG_FUJITSU)
00240 # define COMPILER_ID "FujitsuClang"
00241 # define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00242 # define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00243 # define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00244 # define COMPILER_VERSION_INTERNAL_STR __clang_version__
00245
00246
00247 #elif defined(__FUJITSU)
00248 # define COMPILER_ID "Fujitsu"
00249 # if defined(__FCC_version__)
00250 #   define COMPILER_VERSION __FCC_version__
00251 # elif defined(__FCC_major__)
00252 #   define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00253 #   define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00254 #   define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00255 # endif
00256 # if defined(__fcc_version)
00257 #   define COMPILER_VERSION_INTERNAL DEC(__fcc_version)
00258 # elif defined(__FCC_VERSION)
00259 #   define COMPILER_VERSION_INTERNAL DEC(__FCC_VERSION)
00260 # endif
00261
00262
00263 #elif defined(__ghs__)
00264 # define COMPILER_ID "GHS"
00265 /* __GHS_VERSION_NUMBER = VVVVRP */
00266 # ifdef __GHS_VERSION_NUMBER
00267 # define COMPILER_VERSION_MAJOR DEC(__GHS_VERSION_NUMBER / 100)
00268 # define COMPILER_VERSION_MINOR DEC(__GHS_VERSION_NUMBER / 10 % 10)
00269 # define COMPILER_VERSION_PATCH DEC(__GHS_VERSION_NUMBER      % 10)
00270 # endif
00271
00272 #elif defined(__TASKING__)
00273 # define COMPILER_ID "Tasking"
00274   # define COMPILER_VERSION_MAJOR DEC(__VERSION__/1000)
00275   # define COMPILER_VERSION_MINOR DEC(__VERSION__ % 100)
00276 # define COMPILER_VERSION_INTERNAL DEC(__VERSION__)
00277
00278 #elif defined(__TINYC__)
00279 # define COMPILER_ID "TinyCC"
00280
00281 #elif defined(__BCC__)
00282 # define COMPILER_ID "Bruce"
00283
00284 #elif defined(__SCO_VERSION__)
00285 # define COMPILER_ID "SCO"
00286
00287 #elif defined(__ARMCC_VERSION) && !defined(__clang__)
00288 # define COMPILER_ID "ARMCC"
00289 #if __ARMCC_VERSION >= 1000000
00290   /* __ARMCC_VERSION = VRRPPPP */
```

```
00291   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/1000000)
00292   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 100)
00293   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION     % 10000)
00294 #else
00295   /* __ARMCC_VERSION = VRPPPP */
00296   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/100000)
00297   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 10)
00298   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION    % 10000)
00299 #endif
00300
00301
00302 #elif defined(__clang__) && defined(__apple_build_version__)
00303 # define COMPILER_ID "AppleClang"
00304 # if defined(_MSC_VER)
00305 #  define SIMULATE_ID "MSVC"
00306 # endif
00307 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00308 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00309 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00310 # if defined(_MSC_VER)
00311   /* _MSC_VER = VVRR */
00312 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00313 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00314 # endif
00315 # define COMPILER_VERSION_TWEAK DEC(__apple_build_version__)
00316
00317 #elif defined(__clang__) && defined(__ARMCOMPILER_VERSION)
00318 # define COMPILER_ID "ARMClang"
00319   # define COMPILER_VERSION_MAJOR DEC(__ARMCOMPILER_VERSION/1000000)
00320   # define COMPILER_VERSION_MINOR DEC(__ARMCOMPILER_VERSION/10000 % 100)
00321   # define COMPILER_VERSION_PATCH DEC(__ARMCOMPILER_VERSION     % 10000)
00322 # define COMPILER_VERSION_INTERNAL DEC(__ARMCOMPILER_VERSION)
00323
00324 #elif defined(__clang__)
00325 # define COMPILER_ID "Clang"
00326 # if defined(_MSC_VER)
00327 #  define SIMULATE_ID "MSVC"
00328 # endif
00329 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00330 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00331 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00332 # if defined(_MSC_VER)
00333   /* _MSC_VER = VVRR */
00334 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00335 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00336 # endif
00337
00338 #elif defined(__LCC__) && (defined(__GNUC__) || defined(__GNUG__) || defined(__MCST__))
00339 # define COMPILER_ID "LCC"
00340 # define COMPILER_VERSION_MAJOR DEC(1)
00341 # if defined(__LCC__)
00342 #  define COMPILER_VERSION_MINOR DEC(__LCC__- 100)
00343 # endif
00344 # if defined(__LCC_MINOR__)
00345 #  define COMPILER_VERSION_PATCH DEC(__LCC_MINOR__)
00346 # endif
00347 # if defined(__GNUC__) && defined(__GNUC_MINOR__)
00348 #  define SIMULATE_ID "GNU"
00349 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00350 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00351 #  if defined(__GNUC_PATCHLEVEL__)
00352 #   define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00353 #  endif
00354 # endif
00355
00356 #elif defined(__GNUC__)
00357 # define COMPILER_ID "GNU"
00358 # define COMPILER_VERSION_MAJOR DEC(__GNUC__)
00359 # if defined(__GNUC_MINOR__)
00360 #  define COMPILER_VERSION_MINOR DEC(__GNUC_MINOR__)
00361 # endif
00362 # if defined(__GNUC_PATCHLEVEL__)
00363 #  define COMPILER_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00364 # endif
00365
00366 #elif defined(_MSC_VER)
00367 # define COMPILER_ID "MSVC"
00368   /* _MSC_VER = VVRR */
00369 # define COMPILER_VERSION_MAJOR DEC(_MSC_VER / 100)
00370 # define COMPILER_VERSION_MINOR DEC(_MSC_VER % 100)
00371 # if defined(_MSC_FULL_VER)
00372 #  if _MSC_VER >= 1400
00373     /* _MSC_FULL_VER = VVRRPPPPP */
00374 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 100000)
00375 #  else
00376     /* _MSC_FULL_VER = VVRRPPPP */
00377 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 10000)
```

```
00378 #  endif
00379 # endif
00380 # if defined(_MSC_BUILD)
00381 #  define COMPILER_VERSION_TWEAK DEC(_MSC_BUILD)
00382 # endif
00383
00384 #elif defined(_ADI_COMPILER)
00385 # define COMPILER_ID "ADSP"
00386 #if defined(__VERSIONNUM__)
00387   /* __VERSIONNUM__ = 0xVVRRPPTT */
00388 #  define COMPILER_VERSION_MAJOR DEC(__VERSIONNUM__ » 24 & 0xFF)
00389 #  define COMPILER_VERSION_MINOR DEC(__VERSIONNUM__ » 16 & 0xFF)
00390 #  define COMPILER_VERSION_PATCH DEC(__VERSIONNUM__ » 8 & 0xFF)
00391 #  define COMPILER_VERSION_TWEAK DEC(__VERSIONNUM__ & 0xFF)
00392 #endif
00393
00394 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00395 # define COMPILER_ID "IAR"
00396 # if defined(__VER__) && defined(__ICCARM__)
00397 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 1000000)
00398 #  define COMPILER_VERSION_MINOR DEC(((__VER__) / 1000) % 1000)
00399 #  define COMPILER_VERSION_PATCH DEC((__VER__) % 1000)
00400 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00401 # elif defined(__VER__) && (defined(__ICCAVR__) || defined(__ICCRX__) || defined(__ICCRH850__) ||
      defined(__ICCRL78__) || defined(__ICC430__) || defined(__ICCRISCV__) || defined(__ICCV850__) ||
      defined(__ICC8051__) || defined(__ICCSTM8__))
00402 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 100)
00403 #  define COMPILER_VERSION_MINOR DEC((__VER__) - (((__VER__) / 100)*100))
00404 #  define COMPILER_VERSION_PATCH DEC(__SUBVERSION__)
00405 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00406 # endif
00407
00408 #elif defined(__SDCC_VERSION_MAJOR) || defined(SDCC)
00409 # define COMPILER_ID "SDCC"
00410 # if defined(__SDCC_VERSION_MAJOR)
00411 #  define COMPILER_VERSION_MAJOR DEC(__SDCC_VERSION_MAJOR)
00412 #  define COMPILER_VERSION_MINOR DEC(__SDCC_VERSION_MINOR)
00413 #  define COMPILER_VERSION_PATCH DEC(__SDCC_VERSION_PATCH)
00414 # else
00415   /* SDCC = VRP */
00416 #  define COMPILER_VERSION_MAJOR DEC(SDCC/100)
00417 #  define COMPILER_VERSION_MINOR DEC(SDCC/10 % 10)
00418 #  define COMPILER_VERSION_PATCH DEC(SDCC    % 10)
00419 # endif
00420
00421
00422 /* These compilers are either not known or too old to define an
00423   identification macro.  Try to identify the platform and guess that
00424   it is the native compiler.  */
00425 #elif defined(__hpux) || defined(__hpua)
00426 # define COMPILER_ID "HP"
00427
00428 #else /* unknown compiler */
00429 # define COMPILER_ID ""
00430 #endif
00431
00432 /* Construct the string literal in pieces to prevent the source from
00433    getting matched.  Store it in a pointer rather than an array
00434    because some compilers will just produce instructions to fill the
00435    array rather than assigning a pointer to a static array.  */
00436 char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]";
00437 #ifdef SIMULATE_ID
00438 char const* info_simulate = "INFO" ":" "simulate[" SIMULATE_ID "]";
00439 #endif
00440
00441 #ifdef __QNXNTO__
00442 char const* qnxnto = "INFO" ":" "qnxnto[]";
00443 #endif
00444
00445 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00446 char const *info_cray = "INFO" ":" "compiler_wrapper[CrayPrgEnv]";
00447 #endif
00448
00449 #define STRINGIFY_HELPER(X) #X
00450 #define STRINGIFY(X) STRINGIFY_HELPER(X)
00451
00452 /* Identify known platforms by name.  */
00453 #if defined(__linux) || defined(__linux__) || defined(linux)
00454 # define PLATFORM_ID "Linux"
00455
00456 #elif defined(__MSYS__)
00457 # define PLATFORM_ID "MSYS"
00458
00459 #elif defined(__CYGWIN__)
00460 # define PLATFORM_ID "Cygwin"
00461
00462 #elif defined(__MINGW32__)
```

```
00463 # define PLATFORM_ID "MinGW"
00464
00465 #elif defined(__APPLE__)
00466 # define PLATFORM_ID "Darwin"
00467
00468 #elif defined(_WIN32) || defined(__WIN32__) || defined(WIN32)
00469 # define PLATFORM_ID "Windows"
00470
00471 #elif defined(__FreeBSD__) || defined(__FreeBSD)
00472 # define PLATFORM_ID "FreeBSD"
00473
00474 #elif defined(__NetBSD__) || defined(__NetBSD)
00475 # define PLATFORM_ID "NetBSD"
00476
00477 #elif defined(__OpenBSD__) || defined(__OPENBSD)
00478 # define PLATFORM_ID "OpenBSD"
00479
00480 #elif defined(__sun) || defined(sun)
00481 # define PLATFORM_ID "SunOS"
00482
00483 #elif defined(_AIX) || defined(__AIX) || defined(__AIX__) || defined(__aix) || defined(__aix__)
00484 # define PLATFORM_ID "AIX"
00485
00486 #elif defined(__hpux) || defined(__hpux__)
00487 # define PLATFORM_ID "HP-UX"
00488
00489 #elif defined(__HAIKU__)
00490 # define PLATFORM_ID "Haiku"
00491
00492 #elif defined(__BeOS) || defined(__BEOS__) || defined(_BEOS)
00493 # define PLATFORM_ID "BeOS"
00494
00495 #elif defined(__QNX__) || defined(__QNXNTO__)
00496 # define PLATFORM_ID "QNX"
00497
00498 #elif defined(__tru64) || defined(_tru64) || defined(__TRU64__)
00499 # define PLATFORM_ID "Tru64"
00500
00501 #elif defined(__riscos) || defined(__riscos__)
00502 # define PLATFORM_ID "RISCos"
00503
00504 #elif defined(__sinix) || defined(__sinix__) || defined(__SINIX__)
00505 # define PLATFORM_ID "SINIX"
00506
00507 #elif defined(__UNIX_SV__)
00508 # define PLATFORM_ID "UNIX_SV"
00509
00510 #elif defined(__bsdos__)
00511 # define PLATFORM_ID "BSDOS"
00512
00513 #elif defined(_MPRAS) || defined(MPRAS)
00514 # define PLATFORM_ID "MP-RAS"
00515
00516 #elif defined(__osf) || defined(__osf__)
00517 # define PLATFORM_ID "OSF1"
00518
00519 #elif defined(_SCO_SV) || defined(SCO_SV) || defined(sco_sv)
00520 # define PLATFORM_ID "SCO_SV"
00521
00522 #elif defined(__ultrix) || defined(__ultrix__) || defined(_ULTRIX)
00523 # define PLATFORM_ID "ULTRIX"
00524
00525 #elif defined(__XENIX__) || defined(_XENIX) || defined(XENIX)
00526 # define PLATFORM_ID "Xenix"
00527
00528 #elif defined(__WATCOMC__)
00529 # if defined(__LINUX__)
00530 #  define PLATFORM_ID "Linux"
00531
00532 # elif defined(__DOS__)
00533 #  define PLATFORM_ID "DOS"
00534
00535 # elif defined(__OS2__)
00536 #  define PLATFORM_ID "OS2"
00537
00538 # elif defined(__WINDOWS__)
00539 #  define PLATFORM_ID "Windows3x"
00540
00541 # elif defined(__VXWORKS__)
00542 #  define PLATFORM_ID "VxWorks"
00543
00544 # else /* unknown platform */
00545 #  define PLATFORM_ID
00546 # endif
00547
00548 #elif defined(__INTEGRITY)
00549 # if defined(INT_178B)
```

```
00550 #  define PLATFORM_ID "Integrity178"
00551
00552 # else /* regular Integrity */
00553 #  define PLATFORM_ID "Integrity"
00554 # endif
00555
00556 # elif defined(_ADI_COMPILER)
00557 #  define PLATFORM_ID "ADSP"
00558
00559 #else /* unknown platform */
00560 # define PLATFORM_ID
00561
00562 #endif
00563
00564 /* For windows compilers MSVC and Intel we can determine
00565    the architecture of the compiler being used.  This is because
00566    the compilers do not have flags that can change the architecture,
00567    but rather depend on which compiler is being used
00568 */
00569 #if defined(_WIN32) && defined(_MSC_VER)
00570 # if defined(_M_IA64)
00571 #  define ARCHITECTURE_ID "IA64"
00572
00573 # elif defined(_M_ARM64EC)
00574 #  define ARCHITECTURE_ID "ARM64EC"
00575
00576 # elif defined(_M_X64) || defined(_M_AMD64)
00577 #  define ARCHITECTURE_ID "x64"
00578
00579 # elif defined(_M_IX86)
00580 #  define ARCHITECTURE_ID "X86"
00581
00582 # elif defined(_M_ARM64)
00583 #  define ARCHITECTURE_ID "ARM64"
00584
00585 # elif defined(_M_ARM)
00586 #  if _M_ARM == 4
00587 #   define ARCHITECTURE_ID "ARMV4I"
00588 #  elif _M_ARM == 5
00589 #   define ARCHITECTURE_ID "ARMV5I"
00590 #  else
00591 #   define ARCHITECTURE_ID "ARMV" STRINGIFY(_M_ARM)
00592 #  endif
00593
00594 # elif defined(_M_MIPS)
00595 #  define ARCHITECTURE_ID "MIPS"
00596
00597 # elif defined(_M_SH)
00598 #  define ARCHITECTURE_ID "SHx"
00599
00600 # else /* unknown architecture */
00601 #  define ARCHITECTURE_ID ""
00602 # endif
00603
00604 #elif defined(__WATCOMC__)
00605 # if defined(_M_I86)
00606 #  define ARCHITECTURE_ID "I86"
00607
00608 # elif defined(_M_IX86)
00609 #  define ARCHITECTURE_ID "X86"
00610
00611 # else /* unknown architecture */
00612 #  define ARCHITECTURE_ID ""
00613 # endif
00614
00615 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00616 # if defined(__ICCARM__)
00617 #  define ARCHITECTURE_ID "ARM"
00618
00619 # elif defined(__ICCRX__)
00620 #  define ARCHITECTURE_ID "RX"
00621
00622 # elif defined(__ICCRH850__)
00623 #  define ARCHITECTURE_ID "RH850"
00624
00625 # elif defined(__ICCRL78__)
00626 #  define ARCHITECTURE_ID "RL78"
00627
00628 # elif defined(__ICCRISCV__)
00629 #  define ARCHITECTURE_ID "RISCV"
00630
00631 # elif defined(__ICCAVR__)
00632 #  define ARCHITECTURE_ID "AVR"
00633
00634 # elif defined(__ICC430__)
00635 #  define ARCHITECTURE_ID "MSP430"
00636
```

```
00637 # elif defined(__ICCV850__)
00638 #  define ARCHITECTURE_ID "V850"
00639
00640 # elif defined(__ICC8051__)
00641 #  define ARCHITECTURE_ID "8051"
00642
00643 # elif defined(__ICCSTM8__)
00644 #  define ARCHITECTURE_ID "STM8"
00645
00646 # else /* unknown architecture */
00647 #  define ARCHITECTURE_ID ""
00648 # endif
00649
00650 #elif defined(__ghs__)
00651 # if defined(__PPC64__)
00652 #  define ARCHITECTURE_ID "PPC64"
00653
00654 # elif defined(__ppc__)
00655 #  define ARCHITECTURE_ID "PPC"
00656
00657 # elif defined(__ARM__)
00658 #  define ARCHITECTURE_ID "ARM"
00659
00660 # elif defined(__x86_64__)
00661 #  define ARCHITECTURE_ID "x64"
00662
00663 # elif defined(__i386__)
00664 #  define ARCHITECTURE_ID "X86"
00665
00666 # else /* unknown architecture */
00667 #  define ARCHITECTURE_ID ""
00668 # endif
00669
00670 #elif defined(__TI_COMPILER_VERSION__)
00671 # if defined(__TI_ARM__)
00672 #  define ARCHITECTURE_ID "ARM"
00673
00674 # elif defined(__MSP430__)
00675 #  define ARCHITECTURE_ID "MSP430"
00676
00677 # elif defined(__TMS320C28XX__)
00678 #  define ARCHITECTURE_ID "TMS320C28x"
00679
00680 # elif defined(__TMS320C6X__) || defined(_TMS320C6X)
00681 #  define ARCHITECTURE_ID "TMS320C6x"
00682
00683 # else /* unknown architecture */
00684 #  define ARCHITECTURE_ID ""
00685 # endif
00686
00687 # elif defined(__ADSPSHARC__)
00688 #  define ARCHITECTURE_ID "SHARC"
00689
00690 # elif defined(__ADSPBLACKFIN__)
00691 #  define ARCHITECTURE_ID "Blackfin"
00692
00693 #elif defined(__TASKING__)
00694
00695 # if defined(__CTC__) || defined(__CPTC__)
00696 #  define ARCHITECTURE_ID "TriCore"
00697
00698 # elif defined(__CMCS__)
00699 #  define ARCHITECTURE_ID "MCS"
00700
00701 # elif defined(__CARM__)
00702 #  define ARCHITECTURE_ID "ARM"
00703
00704 # elif defined(__CARC__)
00705 #  define ARCHITECTURE_ID "ARC"
00706
00707 # elif defined(__C51__)
00708 #  define ARCHITECTURE_ID "8051"
00709
00710 # elif defined(__CPCP__)
00711 #  define ARCHITECTURE_ID "PCP"
00712
00713 # else
00714 #  define ARCHITECTURE_ID ""
00715 # endif
00716
00717 #else
00718 #  define ARCHITECTURE_ID
00719 #endif
00720
00721 /* Convert integer to decimal digit literals.  */
00722 #define DEC(n)                   \
00723   ('0' + (((n) / 10000000)%10)), \
```

```
00724   ('0' + (((n) / 1000000)%10)),  \
00725   ('0' + (((n) / 100000)%10)),   \
00726   ('0' + (((n) / 10000)%10)),    \
00727   ('0' + (((n) / 1000)%10)),     \
00728   ('0' + (((n) / 100)%10)),      \
00729   ('0' + (((n) / 10)%10)),       \
00730   ('0' +  ((n) % 10))
00731
00732 /* Convert integer to hex digit literals.  */
00733 #define HEX(n)             \
00734   ('0' + ((n)>>28 & 0xF)), \
00735   ('0' + ((n)>>24 & 0xF)), \
00736   ('0' + ((n)>>20 & 0xF)), \
00737   ('0' + ((n)>>16 & 0xF)), \
00738   ('0' + ((n)>>12 & 0xF)), \
00739   ('0' + ((n)>>8  & 0xF)), \
00740   ('0' + ((n)>>4  & 0xF)), \
00741   ('0' + ((n)     & 0xF))
00742
00743 /* Construct a string literal encoding the version number. */
00744 #ifdef COMPILER_VERSION
00745 char const* info_version = "INFO" ":" "compiler_version[" COMPILER_VERSION "]";
00746
00747 /* Construct a string literal encoding the version number components. */
00748 #elif defined(COMPILER_VERSION_MAJOR)
00749 char const info_version[] = {
00750   'I', 'N', 'F', 'O', ':',
00751   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','[',
00752   COMPILER_VERSION_MAJOR,
00753 # ifdef COMPILER_VERSION_MINOR
00754   '.', COMPILER_VERSION_MINOR,
00755 #  ifdef COMPILER_VERSION_PATCH
00756   '.', COMPILER_VERSION_PATCH,
00757 #   ifdef COMPILER_VERSION_TWEAK
00758     '.', COMPILER_VERSION_TWEAK,
00759 #   endif
00760 #  endif
00761 # endif
00762   ']','\0'};
00763 #endif
00764
00765 /* Construct a string literal encoding the internal version number. */
00766 #ifdef COMPILER_VERSION_INTERNAL
00767 char const info_version_internal[] = {
00768   'I', 'N', 'F', 'O', ':',
00769   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','_',
00770   'i','n','t','e','r','n','a','l','[',
00771   COMPILER_VERSION_INTERNAL,']','\0'};
00772 #elif defined(COMPILER_VERSION_INTERNAL_STR)
00773 char const* info_version_internal = "INFO" ":" "compiler_version_internal["
    COMPILER_VERSION_INTERNAL_STR "]";
00774 #endif
00775
00776 /* Construct a string literal encoding the version number components. */
00777 #ifdef SIMULATE_VERSION_MAJOR
00778 char const info_simulate_version[] = {
00779   'I', 'N', 'F', 'O', ':',
00780   's','i','m','u','l','a','t','e','_','v','e','r','s','i','o','n','[',
00781   SIMULATE_VERSION_MAJOR,
00782 # ifdef SIMULATE_VERSION_MINOR
00783   '.', SIMULATE_VERSION_MINOR,
00784 #  ifdef SIMULATE_VERSION_PATCH
00785   '.', SIMULATE_VERSION_PATCH,
00786 #   ifdef SIMULATE_VERSION_TWEAK
00787     '.', SIMULATE_VERSION_TWEAK,
00788 #   endif
00789 #  endif
00790 # endif
00791   ']','\0'};
00792 #endif
00793
00794 /* Construct the string literal in pieces to prevent the source from
00795    getting matched.  Store it in a pointer rather than an array
00796    because some compilers will just produce instructions to fill the
00797    array rather than assigning a pointer to a static array.  */
00798 char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]";
00799 char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]";
00800
00801
00802
00803 #if !defined(__STDC__) && !defined(__clang__)
00804 # if defined(_MSC_VER) || defined(__ibmxl__) || defined(__IBMC__)
00805 #  define C_VERSION "90"
00806 # else
00807 #  define C_VERSION
00808 # endif
00809 #elif __STDC_VERSION__ > 201710L
```

```
00810 # define C_VERSION "23"
00811 #elif __STDC_VERSION__ >= 201710L
00812 # define C_VERSION "17"
00813 #elif __STDC_VERSION__ >= 201000L
00814 # define C_VERSION "11"
00815 #elif __STDC_VERSION__ >= 199901L
00816 # define C_VERSION "99"
00817 #else
00818 # define C_VERSION "90"
00819 #endif
00820 const char* info_language_standard_default =
00821   "INFO" ":" "standard_default[" C_VERSION "]";
00822
00823 const char* info_language_extensions_default = "INFO" ":" "extensions_default["
00824 #if (defined(__clang__) || defined(__GNUC__) || defined(__xlC__) ||           \
00825      defined(__TI_COMPILER_VERSION__)) &&                                     \
00826   !defined(__STRICT_ANSI__)
00827   "ON"
00828 #else
00829   "OFF"
00830 #endif
00831 "]";
00832
00833 /*--------------------------------------------------------------------------*/
00834
00835 #ifdef ID_VOID_MAIN
00836 void main() {}
00837 #else
00838 # if defined(__CLASSIC_C__)
00839 int main(argc, argv) int argc; char *argv[];
00840 # else
00841 int main(int argc, char* argv[])
00842 # endif
00843 {
00844   int require = 0;
00845   require += info_compiler[argc];
00846   require += info_platform[argc];
00847   require += info_arch[argc];
00848 #ifdef COMPILER_VERSION_MAJOR
00849   require += info_version[argc];
00850 #endif
00851 #ifdef COMPILER_VERSION_INTERNAL
00852   require += info_version_internal[argc];
00853 #endif
00854 #ifdef SIMULATE_ID
00855   require += info_simulate[argc];
00856 #endif
00857 #ifdef SIMULATE_VERSION_MAJOR
00858   require += info_simulate_version[argc];
00859 #endif
00860 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00861   require += info_cray[argc];
00862 #endif
00863   require += info_language_standard_default[argc];
00864   require += info_language_extensions_default[argc];
00865   (void)argv;
00866   return require;
00867 }
00868 #endif
```

6.3 build/default/CMakeFiles/3.25.1/CompilerIdC/CMakeCCompilerId.c File Reference

Macros

- #define __has_include(x) 0
- #define COMPILER_ID ""
- #define STRINGIFY_HELPER(X) #X
- #define STRINGIFY(X) STRINGIFY_HELPER(X)
- #define PLATFORM_ID
- #define ARCHITECTURE_ID
- #define DEC(n)
- #define HEX(n)
- #define C_VERSION

Functions

- int main (int argc, char ∗argv[ ])

Variables

- char const ∗ info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
- char const ∗ info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
- char const ∗ info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
- const char ∗ info_language_standard_default
- const char ∗ info_language_extensions_default

6.3.1 Macro Definition Documentation

6.3.1.1 __has_include

```
#define __has_include(
              x ) 0
```

Definition at line 17 of file CMakeCCompilerId.c.

6.3.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

Definition at line 718 of file CMakeCCompilerId.c.

6.3.1.3 C_VERSION

```
#define C_VERSION
```

Definition at line 807 of file CMakeCCompilerId.c.

6.3.1.4 COMPILER_ID

```
#define COMPILER_ID ""
```

Definition at line 429 of file CMakeCCompilerId.c.

6.3.1.5 DEC

```
#define DEC(
              n )
```

Value:

```
  ('0' + (((n) / 10000000)%10)), \
  ('0' + (((n) / 1000000)%10)),  \
  ('0' + (((n) / 100000)%10)),   \
  ('0' + (((n) / 10000)%10)),    \
  ('0' + (((n) / 1000)%10)),     \
  ('0' + (((n) / 100)%10)),      \
  ('0' + (((n) / 10)%10)),       \
  ('0' +  ((n) % 10))
```

Definition at line 722 of file CMakeCCompilerId.c.

6.3.1.6 HEX

```
#define HEX(
              n )
```

Value:

```
  ('0' + ((n)»28 & 0xF)), \
  ('0' + ((n)»24 & 0xF)), \
  ('0' + ((n)»20 & 0xF)), \
  ('0' + ((n)»16 & 0xF)), \
  ('0' + ((n)»12 & 0xF)), \
  ('0' + ((n)»8  & 0xF)), \
  ('0' + ((n)»4  & 0xF)), \
  ('0' + ((n)     & 0xF))
```

Definition at line 733 of file CMakeCCompilerId.c.

6.3.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```

Definition at line 560 of file CMakeCCompilerId.c.

6.3.1.8 STRINGIFY

```
#define STRINGIFY(
              X ) STRINGIFY_HELPER(X)
```

Definition at line 450 of file CMakeCCompilerId.c.

6.3.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(
              X ) #X
```

Definition at line 449 of file CMakeCCompilerId.c.

6.3.2 Function Documentation

6.3.2.1 main()

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

Definition at line 841 of file CMakeCCompilerId.c.

6.3.3 Variable Documentation

6.3.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

Definition at line 799 of file CMakeCCompilerId.c.

6.3.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

Definition at line 436 of file CMakeCCompilerId.c.

6.3.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["



  "OFF"

"]"
```

Definition at line 823 of file CMakeCCompilerId.c.

6.3.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
=
  "INFO" ":" "standard_default[" C_VERSION "]"
```

Definition at line 820 of file CMakeCCompilerId.c.

6.3.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

Definition at line 798 of file CMakeCCompilerId.c.

6.4 CMakeCCompilerId.c

Go to the documentation of this file.

```
00001 #ifdef __cplusplus
00002 # error "A C++ compiler has been selected for C."
00003 #endif
00004
00005 #if defined(__18CXX)
00006 # define ID_VOID_MAIN
00007 #endif
00008 #if defined(__CLASSIC_C__)
00009 /* cv-qualifiers did not exist in K&R C */
00010 # define const
00011 # define volatile
00012 #endif
00013
00014 #if !defined(__has_include)
00015 /* If the compiler does not have __has_include, pretend the answer is
00016    always no.  */
00017 #  define __has_include(x) 0
00018 #endif
```

```
00019
00020
00021 /* Version number components: V=Version, R=Revision, P=Patch
00022    Version date components:   YYYY=Year, MM=Month,   DD=Day  */
00023
00024 #if defined(__INTEL_COMPILER) || defined(__ICC)
00025 # define COMPILER_ID "Intel"
00026 # if defined(_MSC_VER)
00027 #  define SIMULATE_ID "MSVC"
00028 # endif
00029 # if defined(__GNUC__)
00030 #  define SIMULATE_ID "GNU"
00031 # endif
00032   /* __INTEL_COMPILER = VRP prior to 2021, and then VVVV for 2021 and later,
00033      except that a few beta releases use the old format with V=2021.  */
00034 # if __INTEL_COMPILER < 2021 || __INTEL_COMPILER == 202110 || __INTEL_COMPILER == 202111
00035 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER/100)
00036 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER/10 % 10)
00037 #  if defined(__INTEL_COMPILER_UPDATE)
00038 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER_UPDATE)
00039 #  else
00040 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER   % 10)
00041 #  endif
00042 # else
00043 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER)
00044 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER_UPDATE)
00045    /* The third version component from --version is an update index,
00046       but no macro is provided for it.  */
00047 #  define COMPILER_VERSION_PATCH DEC(0)
00048 # endif
00049 # if defined(__INTEL_COMPILER_BUILD_DATE)
00050    /* __INTEL_COMPILER_BUILD_DATE = YYYYMMDD */
00051 #  define COMPILER_VERSION_TWEAK DEC(__INTEL_COMPILER_BUILD_DATE)
00052 # endif
00053 # if defined(_MSC_VER)
00054    /* _MSC_VER = VVRR */
00055 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00056 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00057 # endif
00058 # if defined(__GNUC__)
00059 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00060 # elif defined(__GNUG__)
00061 #  define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00062 # endif
00063 # if defined(__GNUC_MINOR__)
00064 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00065 # endif
00066 # if defined(__GNUC_PATCHLEVEL__)
00067 #  define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00068 # endif
00069
00070 #elif (defined(__clang__) && defined(__INTEL_CLANG_COMPILER)) || defined(__INTEL_LLVM_COMPILER)
00071 # define COMPILER_ID "IntelLLVM"
00072 #if defined(_MSC_VER)
00073 # define SIMULATE_ID "MSVC"
00074 #endif
00075 #if defined(__GNUC__)
00076 # define SIMULATE_ID "GNU"
00077 #endif
00078 /* __INTEL_LLVM_COMPILER = VVVVRP prior to 2021.2.0, VVVVRRPP for 2021.2.0 and
00079  * later.  Look for 6 digit vs. 8 digit version number to decide encoding.
00080  * VVVV is no smaller than the current year when a version is released.
00081  */
00082 #if __INTEL_LLVM_COMPILER < 1000000L
00083 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/100)
00084 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/10 % 10)
00085 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER    % 10)
00086 #else
00087 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/10000)
00088 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/100 % 100)
00089 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER     % 100)
00090 #endif
00091 #if defined(_MSC_VER)
00092   /* _MSC_VER = VVRR */
00093 # define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00094 # define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00095 #endif
00096 #if defined(__GNUC__)
00097 # define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00098 #elif defined(__GNUG__)
00099 # define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00100 #endif
00101 #if defined(__GNUC_MINOR__)
00102 # define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00103 #endif
00104 #if defined(__GNUC_PATCHLEVEL__)
00105 # define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
```

```
00106 #endif
00107
00108 #elif defined(__PATHCC__)
00109 # define COMPILER_ID "PathScale"
00110 # define COMPILER_VERSION_MAJOR DEC(__PATHCC__)
00111 # define COMPILER_VERSION_MINOR DEC(__PATHCC_MINOR__)
00112 # if defined(__PATHCC_PATCHLEVEL__)
00113 #  define COMPILER_VERSION_PATCH DEC(__PATHCC_PATCHLEVEL__)
00114 # endif
00115
00116 #elif defined(__BORLANDC__) && defined(__CODEGEARC_VERSION__)
00117 # define COMPILER_ID "Embarcadero"
00118 # define COMPILER_VERSION_MAJOR HEX(__CODEGEARC_VERSION__»24 & 0x00FF)
00119 # define COMPILER_VERSION_MINOR HEX(__CODEGEARC_VERSION__»16 & 0x00FF)
00120 # define COMPILER_VERSION_PATCH DEC(__CODEGEARC_VERSION__     & 0xFFFF)
00121
00122 #elif defined(__BORLANDC__)
00123 # define COMPILER_ID "Borland"
00124   /* __BORLANDC__ = 0xVRR */
00125 # define COMPILER_VERSION_MAJOR HEX(__BORLANDC__»8)
00126 # define COMPILER_VERSION_MINOR HEX(__BORLANDC__ & 0xFF)
00127
00128 #elif defined(__WATCOMC__) && __WATCOMC__ < 1200
00129 # define COMPILER_ID "Watcom"
00130    /* __WATCOMC__ = VVRR */
00131 # define COMPILER_VERSION_MAJOR DEC(__WATCOMC__ / 100)
00132 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00133 # if (__WATCOMC__ % 10) > 0
00134 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00135 # endif
00136
00137 #elif defined(__WATCOMC__)
00138 # define COMPILER_ID "OpenWatcom"
00139    /* __WATCOMC__ = VVRP + 1100 */
00140 # define COMPILER_VERSION_MAJOR DEC((__WATCOMC__ - 1100) / 100)
00141 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00142 # if (__WATCOMC__ % 10) > 0
00143 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00144 # endif
00145
00146 #elif defined(__SUNPRO_C)
00147 # define COMPILER_ID "SunPro"
00148 # if __SUNPRO_C >= 0x5100
00149    /* __SUNPRO_C = 0xVRRP */
00150 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_C»12)
00151 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_C»4 & 0xFF)
00152 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_C    & 0xF)
00153 # else
00154    /* __SUNPRO_CC = 0xVRP */
00155 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_C»8)
00156 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_C»4 & 0xF)
00157 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_C    & 0xF)
00158 # endif
00159
00160 #elif defined(__HP_cc)
00161 # define COMPILER_ID "HP"
00162   /* __HP_cc = VVRRPP */
00163 # define COMPILER_VERSION_MAJOR DEC(__HP_cc/10000)
00164 # define COMPILER_VERSION_MINOR DEC(__HP_cc/100 % 100)
00165 # define COMPILER_VERSION_PATCH DEC(__HP_cc     % 100)
00166
00167 #elif defined(__DECC)
00168 # define COMPILER_ID "Compaq"
00169   /* __DECC_VER = VVRRTPPPP */
00170 # define COMPILER_VERSION_MAJOR DEC(__DECC_VER/10000000)
00171 # define COMPILER_VERSION_MINOR DEC(__DECC_VER/100000  % 100)
00172 # define COMPILER_VERSION_PATCH DEC(__DECC_VER         % 10000)
00173
00174 #elif defined(__IBMC__) && defined(__COMPILER_VER__)
00175 # define COMPILER_ID "zOS"
00176   /* __IBMC__ = VRP */
00177 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00178 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00179 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00180
00181 #elif defined(__open_xl__) && defined(__clang__)
00182 # define COMPILER_ID "IBMClang"
00183 # define COMPILER_VERSION_MAJOR DEC(__open_xl_version__)
00184 # define COMPILER_VERSION_MINOR DEC(__open_xl_release__)
00185 # define COMPILER_VERSION_PATCH DEC(__open_xl_modification__)
00186 # define COMPILER_VERSION_TWEAK DEC(__open_xl_ptf_fix_level__)
00187
00188
00189 #elif defined(__ibmxl__) && defined(__clang__)
00190 # define COMPILER_ID "XLClang"
00191 # define COMPILER_VERSION_MAJOR DEC(__ibmxl_version__)
00192 # define COMPILER_VERSION_MINOR DEC(__ibmxl_release__)
```

```
00193 # define COMPILER_VERSION_PATCH DEC(__ibmxl_modification__)
00194 # define COMPILER_VERSION_TWEAK DEC(__ibmxl_ptf_fix_level__)
00195
00196
00197 #elif defined(__IBMC__) && !defined(__COMPILER_VER__) && __IBMC__ >= 800
00198 # define COMPILER_ID "XL"
00199   /* __IBMC__ = VRP */
00200 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00201 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00202 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00203
00204 #elif defined(__IBMC__) && !defined(__COMPILER_VER__) && __IBMC__ < 800
00205 # define COMPILER_ID "VisualAge"
00206   /* __IBMC__ = VRP */
00207 # define COMPILER_VERSION_MAJOR DEC(__IBMC__/100)
00208 # define COMPILER_VERSION_MINOR DEC(__IBMC__/10 % 10)
00209 # define COMPILER_VERSION_PATCH DEC(__IBMC__    % 10)
00210
00211 #elif defined(__NVCOMPILER)
00212 # define COMPILER_ID "NVHPC"
00213 # define COMPILER_VERSION_MAJOR DEC(__NVCOMPILER_MAJOR__)
00214 # define COMPILER_VERSION_MINOR DEC(__NVCOMPILER_MINOR__)
00215 # if defined(__NVCOMPILER_PATCHLEVEL__)
00216 #  define COMPILER_VERSION_PATCH DEC(__NVCOMPILER_PATCHLEVEL__)
00217 # endif
00218
00219 #elif defined(__PGI)
00220 # define COMPILER_ID "PGI"
00221 # define COMPILER_VERSION_MAJOR DEC(__PGIC__)
00222 # define COMPILER_VERSION_MINOR DEC(__PGIC_MINOR__)
00223 # if defined(__PGIC_PATCHLEVEL__)
00224 #  define COMPILER_VERSION_PATCH DEC(__PGIC_PATCHLEVEL__)
00225 # endif
00226
00227 #elif defined(_CRAYC)
00228 # define COMPILER_ID "Cray"
00229 # define COMPILER_VERSION_MAJOR DEC(_RELEASE_MAJOR)
00230 # define COMPILER_VERSION_MINOR DEC(_RELEASE_MINOR)
00231
00232 #elif defined(__TI_COMPILER_VERSION__)
00233 # define COMPILER_ID "TI"
00234   /* __TI_COMPILER_VERSION__ = VVVRRRPPP */
00235 # define COMPILER_VERSION_MAJOR DEC(__TI_COMPILER_VERSION__/1000000)
00236 # define COMPILER_VERSION_MINOR DEC(__TI_COMPILER_VERSION__/1000   % 1000)
00237 # define COMPILER_VERSION_PATCH DEC(__TI_COMPILER_VERSION__        % 1000)
00238
00239 #elif defined(__CLANG_FUJITSU)
00240 # define COMPILER_ID "FujitsuClang"
00241 # define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00242 # define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00243 # define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00244 # define COMPILER_VERSION_INTERNAL_STR __clang_version__
00245
00246
00247 #elif defined(__FUJITSU)
00248 # define COMPILER_ID "Fujitsu"
00249 # if defined(__FCC_version__)
00250 #   define COMPILER_VERSION __FCC_version__
00251 # elif defined(__FCC_major__)
00252 #   define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00253 #   define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00254 #   define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00255 # endif
00256 # if defined(__fcc_version)
00257 #   define COMPILER_VERSION_INTERNAL DEC(__fcc_version)
00258 # elif defined(__FCC_VERSION)
00259 #   define COMPILER_VERSION_INTERNAL DEC(__FCC_VERSION)
00260 # endif
00261
00262
00263 #elif defined(__ghs__)
00264 # define COMPILER_ID "GHS"
00265 /* __GHS_VERSION_NUMBER = VVVVRP */
00266 # ifdef __GHS_VERSION_NUMBER
00267 # define COMPILER_VERSION_MAJOR DEC(__GHS_VERSION_NUMBER / 100)
00268 # define COMPILER_VERSION_MINOR DEC(__GHS_VERSION_NUMBER / 10 % 10)
00269 # define COMPILER_VERSION_PATCH DEC(__GHS_VERSION_NUMBER      % 10)
00270 # endif
00271
00272 #elif defined(__TASKING__)
00273 # define COMPILER_ID "Tasking"
00274   # define COMPILER_VERSION_MAJOR DEC(__VERSION__/1000)
00275   # define COMPILER_VERSION_MINOR DEC(__VERSION__ % 100)
00276 # define COMPILER_VERSION_INTERNAL DEC(__VERSION__)
00277
00278 #elif defined(__TINYC__)
00279 # define COMPILER_ID "TinyCC"
```

```
00280
00281 #elif defined(__BCC__)
00282 # define COMPILER_ID "Bruce"
00283
00284 #elif defined(__SCO_VERSION__)
00285 # define COMPILER_ID "SCO"
00286
00287 #elif defined(__ARMCC_VERSION) && !defined(__clang__)
00288 # define COMPILER_ID "ARMCC"
00289 #if __ARMCC_VERSION >= 1000000
00290   /* __ARMCC_VERSION = VRRPPPP */
00291   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/1000000)
00292   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 100)
00293   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION     % 10000)
00294 #else
00295   /* __ARMCC_VERSION = VRPPPP */
00296   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/100000)
00297   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 10)
00298   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION    % 10000)
00299 #endif
00300
00301
00302 #elif defined(__clang__) && defined(__apple_build_version__)
00303 # define COMPILER_ID "AppleClang"
00304 # if defined(_MSC_VER)
00305 #  define SIMULATE_ID "MSVC"
00306 # endif
00307 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00308 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00309 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00310 # if defined(_MSC_VER)
00311   /* _MSC_VER = VVRR */
00312 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00313 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00314 # endif
00315 # define COMPILER_VERSION_TWEAK DEC(__apple_build_version__)
00316
00317 #elif defined(__clang__) && defined(__ARMCOMPILER_VERSION)
00318 # define COMPILER_ID "ARMClang"
00319   # define COMPILER_VERSION_MAJOR DEC(__ARMCOMPILER_VERSION/1000000)
00320   # define COMPILER_VERSION_MINOR DEC(__ARMCOMPILER_VERSION/10000 % 100)
00321   # define COMPILER_VERSION_PATCH DEC(__ARMCOMPILER_VERSION     % 10000)
00322 # define COMPILER_VERSION_INTERNAL DEC(__ARMCOMPILER_VERSION)
00323
00324 #elif defined(__clang__)
00325 # define COMPILER_ID "Clang"
00326 # if defined(_MSC_VER)
00327 #  define SIMULATE_ID "MSVC"
00328 # endif
00329 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00330 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00331 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00332 # if defined(_MSC_VER)
00333   /* _MSC_VER = VVRR */
00334 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00335 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00336 # endif
00337
00338 #elif defined(__LCC__) && (defined(__GNUC__) || defined(__GNUG__) || defined(__MCST__))
00339 # define COMPILER_ID "LCC"
00340 # define COMPILER_VERSION_MAJOR DEC(1)
00341 # if defined(__LCC__)
00342 #  define COMPILER_VERSION_MINOR DEC(__LCC__- 100)
00343 # endif
00344 # if defined(__LCC_MINOR__)
00345 #  define COMPILER_VERSION_PATCH DEC(__LCC_MINOR__)
00346 # endif
00347 # if defined(__GNUC__) && defined(__GNUC_MINOR__)
00348 #  define SIMULATE_ID "GNU"
00349 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00350 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00351 #  if defined(__GNUC_PATCHLEVEL__)
00352 #   define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00353 #  endif
00354 # endif
00355
00356 #elif defined(__GNUC__)
00357 # define COMPILER_ID "GNU"
00358 # define COMPILER_VERSION_MAJOR DEC(__GNUC__)
00359 # if defined(__GNUC_MINOR__)
00360 #  define COMPILER_VERSION_MINOR DEC(__GNUC_MINOR__)
00361 # endif
00362 # if defined(__GNUC_PATCHLEVEL__)
00363 #  define COMPILER_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00364 # endif
00365
00366 #elif defined(_MSC_VER)
```

```
00367 # define COMPILER_ID "MSVC"
00368   /* _MSC_VER = VVRR */
00369 # define COMPILER_VERSION_MAJOR DEC(_MSC_VER / 100)
00370 # define COMPILER_VERSION_MINOR DEC(_MSC_VER % 100)
00371 # if defined(_MSC_FULL_VER)
00372 #  if _MSC_VER >= 1400
00373     /* _MSC_FULL_VER = VVRRPPPPP */
00374 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 100000)
00375 #  else
00376     /* _MSC_FULL_VER = VVRRPPPP */
00377 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 10000)
00378 #  endif
00379 # endif
00380 # if defined(_MSC_BUILD)
00381 #  define COMPILER_VERSION_TWEAK DEC(_MSC_BUILD)
00382 # endif
00383
00384 #elif defined(_ADI_COMPILER)
00385 # define COMPILER_ID "ADSP"
00386 #if defined(__VERSIONNUM__)
00387   /* __VERSIONNUM__ = 0xVVRRPPTT */
00388 #  define COMPILER_VERSION_MAJOR DEC(__VERSIONNUM__ >> 24 & 0xFF)
00389 #  define COMPILER_VERSION_MINOR DEC(__VERSIONNUM__ >> 16 & 0xFF)
00390 #  define COMPILER_VERSION_PATCH DEC(__VERSIONNUM__ >> 8 & 0xFF)
00391 #  define COMPILER_VERSION_TWEAK DEC(__VERSIONNUM__ & 0xFF)
00392 #endif
00393
00394 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00395 # define COMPILER_ID "IAR"
00396 # if defined(__VER__) && defined(__ICCARM__)
00397 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 1000000)
00398 #  define COMPILER_VERSION_MINOR DEC(((__VER__) / 1000) % 1000)
00399 #  define COMPILER_VERSION_PATCH DEC((__VER__) % 1000)
00400 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00401 # elif defined(__VER__) && (defined(__ICCAVR__) || defined(__ICCRX__) || defined(__ICCRH850__) ||
      defined(__ICCRL78__) || defined(__ICC430__) || defined(__ICCRISCV__) || defined(__ICCV850__) ||
      defined(__ICC8051__) || defined(__ICCSTM8__))
00402 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 100)
00403 #  define COMPILER_VERSION_MINOR DEC((__VER__) - (((__VER__) / 100)*100))
00404 #  define COMPILER_VERSION_PATCH DEC(__SUBVERSION__)
00405 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00406 # endif
00407
00408 #elif defined(__SDCC_VERSION_MAJOR) || defined(SDCC)
00409 # define COMPILER_ID "SDCC"
00410 # if defined(__SDCC_VERSION_MAJOR)
00411 #  define COMPILER_VERSION_MAJOR DEC(__SDCC_VERSION_MAJOR)
00412 #  define COMPILER_VERSION_MINOR DEC(__SDCC_VERSION_MINOR)
00413 #  define COMPILER_VERSION_PATCH DEC(__SDCC_VERSION_PATCH)
00414 # else
00415   /* SDCC = VRP */
00416 #  define COMPILER_VERSION_MAJOR DEC(SDCC/100)
00417 #  define COMPILER_VERSION_MINOR DEC(SDCC/10 % 10)
00418 #  define COMPILER_VERSION_PATCH DEC(SDCC    % 10)
00419 # endif
00420
00421
00422 /* These compilers are either not known or too old to define an
00423   identification macro.  Try to identify the platform and guess that
00424   it is the native compiler.  */
00425 #elif defined(__hpux) || defined(__hpua)
00426 # define COMPILER_ID "HP"
00427
00428 #else /* unknown compiler */
00429 # define COMPILER_ID ""
00430 #endif
00431
00432 /* Construct the string literal in pieces to prevent the source from
00433    getting matched.  Store it in a pointer rather than an array
00434    because some compilers will just produce instructions to fill the
00435    array rather than assigning a pointer to a static array.  */
00436 char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]";
00437 #ifdef SIMULATE_ID
00438 char const* info_simulate = "INFO" ":" "simulate[" SIMULATE_ID "]";
00439 #endif
00440
00441 #ifdef __QNXNTO__
00442 char const* qnxnto = "INFO" ":" "qnxnto[]";
00443 #endif
00444
00445 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00446 char const *info_cray = "INFO" ":" "compiler_wrapper[CrayPrgEnv]";
00447 #endif
00448
00449 #define STRINGIFY_HELPER(X) #X
00450 #define STRINGIFY(X) STRINGIFY_HELPER(X)
00451
```

```
00452 /* Identify known platforms by name.  */
00453 #if defined(__linux) || defined(__linux__) || defined(linux)
00454 # define PLATFORM_ID "Linux"
00455
00456 #elif defined(__MSYS__)
00457 # define PLATFORM_ID "MSYS"
00458
00459 #elif defined(__CYGWIN__)
00460 # define PLATFORM_ID "Cygwin"
00461
00462 #elif defined(__MINGW32__)
00463 # define PLATFORM_ID "MinGW"
00464
00465 #elif defined(__APPLE__)
00466 # define PLATFORM_ID "Darwin"
00467
00468 #elif defined(_WIN32) || defined(__WIN32__) || defined(WIN32)
00469 # define PLATFORM_ID "Windows"
00470
00471 #elif defined(__FreeBSD__) || defined(__FreeBSD)
00472 # define PLATFORM_ID "FreeBSD"
00473
00474 #elif defined(__NetBSD__) || defined(__NetBSD)
00475 # define PLATFORM_ID "NetBSD"
00476
00477 #elif defined(__OpenBSD__) || defined(__OPENBSD)
00478 # define PLATFORM_ID "OpenBSD"
00479
00480 #elif defined(__sun) || defined(sun)
00481 # define PLATFORM_ID "SunOS"
00482
00483 #elif defined(_AIX) || defined(__AIX) || defined(__AIX__) || defined(__aix) || defined(__aix__)
00484 # define PLATFORM_ID "AIX"
00485
00486 #elif defined(__hpux) || defined(__hpux__)
00487 # define PLATFORM_ID "HP-UX"
00488
00489 #elif defined(__HAIKU__)
00490 # define PLATFORM_ID "Haiku"
00491
00492 #elif defined(__BeOS) || defined(__BEOS__) || defined(_BEOS)
00493 # define PLATFORM_ID "BeOS"
00494
00495 #elif defined(__QNX__) || defined(__QNXNTO__)
00496 # define PLATFORM_ID "QNX"
00497
00498 #elif defined(__tru64) || defined(_tru64) || defined(__TRU64__)
00499 # define PLATFORM_ID "Tru64"
00500
00501 #elif defined(__riscos) || defined(__riscos__)
00502 # define PLATFORM_ID "RISCos"
00503
00504 #elif defined(__sinix) || defined(__sinix__) || defined(__SINIX__)
00505 # define PLATFORM_ID "SINIX"
00506
00507 #elif defined(__UNIX_SV__)
00508 # define PLATFORM_ID "UNIX_SV"
00509
00510 #elif defined(__bsdos__)
00511 # define PLATFORM_ID "BSDOS"
00512
00513 #elif defined(_MPRAS) || defined(MPRAS)
00514 # define PLATFORM_ID "MP-RAS"
00515
00516 #elif defined(__osf) || defined(__osf__)
00517 # define PLATFORM_ID "OSF1"
00518
00519 #elif defined(_SCO_SV) || defined(SCO_SV) || defined(sco_sv)
00520 # define PLATFORM_ID "SCO_SV"
00521
00522 #elif defined(__ultrix) || defined(__ultrix__) || defined(_ULTRIX)
00523 # define PLATFORM_ID "ULTRIX"
00524
00525 #elif defined(__XENIX__) || defined(_XENIX) || defined(XENIX)
00526 # define PLATFORM_ID "Xenix"
00527
00528 #elif defined(__WATCOMC__)
00529 # if defined(__LINUX__)
00530 #  define PLATFORM_ID "Linux"
00531
00532 # elif defined(__DOS__)
00533 #  define PLATFORM_ID "DOS"
00534
00535 # elif defined(__OS2__)
00536 #  define PLATFORM_ID "OS2"
00537
00538 # elif defined(__WINDOWS__)
```

```
00539 #  define PLATFORM_ID "Windows3x"
00540
00541 # elif defined(__VXWORKS__)
00542 #  define PLATFORM_ID "VxWorks"
00543
00544 # else /* unknown platform */
00545 #  define PLATFORM_ID
00546 # endif
00547
00548 #elif defined(__INTEGRITY)
00549 # if defined(INT_178B)
00550 #  define PLATFORM_ID "Integrity178"
00551
00552 # else /* regular Integrity */
00553 #  define PLATFORM_ID "Integrity"
00554 # endif
00555
00556 # elif defined(_ADI_COMPILER)
00557 #  define PLATFORM_ID "ADSP"
00558
00559 #else /* unknown platform */
00560 # define PLATFORM_ID
00561
00562 #endif
00563
00564 /* For windows compilers MSVC and Intel we can determine
00565    the architecture of the compiler being used.  This is because
00566    the compilers do not have flags that can change the architecture,
00567    but rather depend on which compiler is being used
00568 */
00569 #if defined(_WIN32) && defined(_MSC_VER)
00570 # if defined(_M_IA64)
00571 #  define ARCHITECTURE_ID "IA64"
00572
00573 # elif defined(_M_ARM64EC)
00574 #  define ARCHITECTURE_ID "ARM64EC"
00575
00576 # elif defined(_M_X64) || defined(_M_AMD64)
00577 #  define ARCHITECTURE_ID "x64"
00578
00579 # elif defined(_M_IX86)
00580 #  define ARCHITECTURE_ID "X86"
00581
00582 # elif defined(_M_ARM64)
00583 #  define ARCHITECTURE_ID "ARM64"
00584
00585 # elif defined(_M_ARM)
00586 #  if _M_ARM == 4
00587 #   define ARCHITECTURE_ID "ARMV4I"
00588 #  elif _M_ARM == 5
00589 #   define ARCHITECTURE_ID "ARMV5I"
00590 #  else
00591 #   define ARCHITECTURE_ID "ARMV" STRINGIFY(_M_ARM)
00592 #  endif
00593
00594 # elif defined(_M_MIPS)
00595 #  define ARCHITECTURE_ID "MIPS"
00596
00597 # elif defined(_M_SH)
00598 #  define ARCHITECTURE_ID "SHx"
00599
00600 # else /* unknown architecture */
00601 #  define ARCHITECTURE_ID ""
00602 # endif
00603
00604 #elif defined(__WATCOMC__)
00605 # if defined(_M_I86)
00606 #  define ARCHITECTURE_ID "I86"
00607
00608 # elif defined(_M_IX86)
00609 #  define ARCHITECTURE_ID "X86"
00610
00611 # else /* unknown architecture */
00612 #  define ARCHITECTURE_ID ""
00613 # endif
00614
00615 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00616 # if defined(__ICCARM__)
00617 #  define ARCHITECTURE_ID "ARM"
00618
00619 # elif defined(__ICCRX__)
00620 #  define ARCHITECTURE_ID "RX"
00621
00622 # elif defined(__ICCRH850__)
00623 #  define ARCHITECTURE_ID "RH850"
00624
00625 # elif defined(__ICCRL78__)
```

```
00626 #  define ARCHITECTURE_ID "RL78"
00627
00628 # elif defined(__ICCRISCV__)
00629 #  define ARCHITECTURE_ID "RISCV"
00630
00631 # elif defined(__ICCAVR__)
00632 #  define ARCHITECTURE_ID "AVR"
00633
00634 # elif defined(__ICC430__)
00635 #  define ARCHITECTURE_ID "MSP430"
00636
00637 # elif defined(__ICCV850__)
00638 #  define ARCHITECTURE_ID "V850"
00639
00640 # elif defined(__ICC8051__)
00641 #  define ARCHITECTURE_ID "8051"
00642
00643 # elif defined(__ICCSTM8__)
00644 #  define ARCHITECTURE_ID "STM8"
00645
00646 # else /* unknown architecture */
00647 #  define ARCHITECTURE_ID ""
00648 # endif
00649
00650 #elif defined(__ghs__)
00651 # if defined(__PPC64__)
00652 #  define ARCHITECTURE_ID "PPC64"
00653
00654 # elif defined(__ppc__)
00655 #  define ARCHITECTURE_ID "PPC"
00656
00657 # elif defined(__ARM__)
00658 #  define ARCHITECTURE_ID "ARM"
00659
00660 # elif defined(__x86_64__)
00661 #  define ARCHITECTURE_ID "x64"
00662
00663 # elif defined(__i386__)
00664 #  define ARCHITECTURE_ID "X86"
00665
00666 # else /* unknown architecture */
00667 #  define ARCHITECTURE_ID ""
00668 # endif
00669
00670 #elif defined(__TI_COMPILER_VERSION__)
00671 # if defined(__TI_ARM__)
00672 #  define ARCHITECTURE_ID "ARM"
00673
00674 # elif defined(__MSP430__)
00675 #  define ARCHITECTURE_ID "MSP430"
00676
00677 # elif defined(__TMS320C28XX__)
00678 #  define ARCHITECTURE_ID "TMS320C28x"
00679
00680 # elif defined(__TMS320C6X__) || defined(_TMS320C6X)
00681 #  define ARCHITECTURE_ID "TMS320C6x"
00682
00683 # else /* unknown architecture */
00684 #  define ARCHITECTURE_ID ""
00685 # endif
00686
00687 # elif defined(__ADSPSHARC__)
00688 #  define ARCHITECTURE_ID "SHARC"
00689
00690 # elif defined(__ADSPBLACKFIN__)
00691 #  define ARCHITECTURE_ID "Blackfin"
00692
00693 #elif defined(__TASKING__)
00694
00695 # if defined(__CTC__) || defined(__CPTC__)
00696 #  define ARCHITECTURE_ID "TriCore"
00697
00698 # elif defined(__CMCS__)
00699 #  define ARCHITECTURE_ID "MCS"
00700
00701 # elif defined(__CARM__)
00702 #  define ARCHITECTURE_ID "ARM"
00703
00704 # elif defined(__CARC__)
00705 #  define ARCHITECTURE_ID "ARC"
00706
00707 # elif defined(__C51__)
00708 #  define ARCHITECTURE_ID "8051"
00709
00710 # elif defined(__CPCP__)
00711 #  define ARCHITECTURE_ID "PCP"
00712
```

```
00713 # else
00714 #  define ARCHITECTURE_ID ""
00715 # endif
00716
00717 #else
00718 #  define ARCHITECTURE_ID
00719 #endif
00720
00721 /* Convert integer to decimal digit literals.  */
00722 #define DEC(n)                   \
00723   ('0' + (((n) / 10000000)%10)), \
00724   ('0' + (((n) / 1000000)%10)),  \
00725   ('0' + (((n) / 100000)%10)),   \
00726   ('0' + (((n) / 10000)%10)),    \
00727   ('0' + (((n) / 1000)%10)),     \
00728   ('0' + (((n) / 100)%10)),      \
00729   ('0' + (((n) / 10)%10)),       \
00730   ('0' +  ((n) % 10))
00731
00732 /* Convert integer to hex digit literals.  */
00733 #define HEX(n)             \
00734   ('0' + ((n)>>28 & 0xF)), \
00735   ('0' + ((n)>>24 & 0xF)), \
00736   ('0' + ((n)>>20 & 0xF)), \
00737   ('0' + ((n)>>16 & 0xF)), \
00738   ('0' + ((n)>>12 & 0xF)), \
00739   ('0' + ((n)>>8  & 0xF)), \
00740   ('0' + ((n)>>4  & 0xF)), \
00741   ('0' + ((n)     & 0xF))
00742
00743 /* Construct a string literal encoding the version number. */
00744 #ifdef COMPILER_VERSION
00745 char const* info_version = "INFO" ":" "compiler_version[" COMPILER_VERSION "]";
00746
00747 /* Construct a string literal encoding the version number components. */
00748 #elif defined(COMPILER_VERSION_MAJOR)
00749 char const info_version[] = {
00750   'I', 'N', 'F', 'O', ':',
00751   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','[',
00752   COMPILER_VERSION_MAJOR,
00753 # ifdef COMPILER_VERSION_MINOR
00754   '.', COMPILER_VERSION_MINOR,
00755 #  ifdef COMPILER_VERSION_PATCH
00756   '.', COMPILER_VERSION_PATCH,
00757 #   ifdef COMPILER_VERSION_TWEAK
00758     '.', COMPILER_VERSION_TWEAK,
00759 #   endif
00760 #  endif
00761 # endif
00762   ']','\0'};
00763 #endif
00764
00765 /* Construct a string literal encoding the internal version number. */
00766 #ifdef COMPILER_VERSION_INTERNAL
00767 char const info_version_internal[] = {
00768   'I', 'N', 'F', 'O', ':',
00769   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','_',
00770   'i','n','t','e','r','n','a','l','[',
00771   COMPILER_VERSION_INTERNAL,']','\0'};
00772 #elif defined(COMPILER_VERSION_INTERNAL_STR)
00773 char const* info_version_internal = "INFO" ":" "compiler_version_internal["
      COMPILER_VERSION_INTERNAL_STR "]";
00774 #endif
00775
00776 /* Construct a string literal encoding the version number components. */
00777 #ifdef SIMULATE_VERSION_MAJOR
00778 char const info_simulate_version[] = {
00779   'I', 'N', 'F', 'O', ':',
00780   's','i','m','u','l','a','t','e','_','v','e','r','s','i','o','n','[',
00781   SIMULATE_VERSION_MAJOR,
00782 # ifdef SIMULATE_VERSION_MINOR
00783   '.', SIMULATE_VERSION_MINOR,
00784 #  ifdef SIMULATE_VERSION_PATCH
00785   '.', SIMULATE_VERSION_PATCH,
00786 #   ifdef SIMULATE_VERSION_TWEAK
00787     '.', SIMULATE_VERSION_TWEAK,
00788 #   endif
00789 #  endif
00790 # endif
00791   ']','\0'};
00792 #endif
00793
00794 /* Construct the string literal in pieces to prevent the source from
00795    getting matched.  Store it in a pointer rather than an array
00796    because some compilers will just produce instructions to fill the
00797    array rather than assigning a pointer to a static array.  */
00798 char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]";
```

```
00799 char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]";
00800
00801
00802
00803 #if !defined(__STDC__) && !defined(__clang__)
00804 # if defined(_MSC_VER) || defined(__ibmxl__) || defined(__IBMC__)
00805 #  define C_VERSION "90"
00806 # else
00807 #  define C_VERSION
00808 # endif
00809 #elif __STDC_VERSION__ > 201710L
00810 # define C_VERSION "23"
00811 #elif __STDC_VERSION__ >= 201710L
00812 # define C_VERSION "17"
00813 #elif __STDC_VERSION__ >= 201000L
00814 # define C_VERSION "11"
00815 #elif __STDC_VERSION__ >= 199901L
00816 # define C_VERSION "99"
00817 #else
00818 # define C_VERSION "90"
00819 #endif
00820 const char* info_language_standard_default =
00821   "INFO" ":" "standard_default[" C_VERSION "]";
00822
00823 const char* info_language_extensions_default = "INFO" ":" "extensions_default["
00824 #if (defined(__clang__) || defined(__GNUC__) || defined(__xlC__) ||           \
00825      defined(__TI_COMPILER_VERSION__)) &&                                     \
00826   !defined(__STRICT_ANSI__)
00827   "ON"
00828 #else
00829   "OFF"
00830 #endif
00831 "]";
00832
00833 /*--------------------------------------------------------------------------*/
00834
00835 #ifdef ID_VOID_MAIN
00836 void main() {}
00837 #else
00838 # if defined(__CLASSIC_C__)
00839 int main(argc, argv) int argc; char *argv[];
00840 # else
00841 int main(int argc, char* argv[])
00842 # endif
00843 {
00844   int require = 0;
00845   require += info_compiler[argc];
00846   require += info_platform[argc];
00847   require += info_arch[argc];
00848 #ifdef COMPILER_VERSION_MAJOR
00849   require += info_version[argc];
00850 #endif
00851 #ifdef COMPILER_VERSION_INTERNAL
00852   require += info_version_internal[argc];
00853 #endif
00854 #ifdef SIMULATE_ID
00855   require += info_simulate[argc];
00856 #endif
00857 #ifdef SIMULATE_VERSION_MAJOR
00858   require += info_simulate_version[argc];
00859 #endif
00860 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00861   require += info_cray[argc];
00862 #endif
00863   require += info_language_standard_default[argc];
00864   require += info_language_extensions_default[argc];
00865   (void)argv;
00866   return require;
00867 }
00868 #endif
```

6.5 build/cmake.debug.linux.x86_64/CMakeFiles/3.25.1/CompilerIdCXX/↩ CMakeCXXCompilerId.cpp File Reference

Macros

- #define __has_include(x) 0
- #define COMPILER_ID ""

- #define STRINGIFY_HELPER(X) #X
- #define STRINGIFY(X) STRINGIFY_HELPER(X)
- #define PLATFORM_ID
- #define ARCHITECTURE_ID
- #define DEC(n)
- #define HEX(n)
- #define CXX_STD __cplusplus

Functions

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

Variables

- char const * info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
- char const * info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
- char const * info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
- const char * info_language_standard_default
- const char * info_language_extensions_default

6.5.1 Macro Definition Documentation

6.5.1.1 __has_include

```
#define __has_include(
              x ) 0
```

Definition at line 11 of file CMakeCXXCompilerId.cpp.

6.5.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

Definition at line 703 of file CMakeCXXCompilerId.cpp.

6.5.1.3 COMPILER_ID

```
#define COMPILER_ID ""
```

Definition at line 414 of file CMakeCXXCompilerId.cpp.

6.5.1.4 CXX_STD

```
#define CXX_STD __cplusplus
```

Definition at line 801 of file CMakeCXXCompilerId.cpp.

6.5.1.5 DEC

```
#define DEC(
              n )
```

Value:

```
  ('0' + (((n) / 10000000)%10)), \
  ('0' + (((n) / 1000000)%10)),  \
  ('0' + (((n) / 100000)%10)),   \
  ('0' + (((n) / 10000)%10)),    \
  ('0' + (((n) / 1000)%10)),     \
  ('0' + (((n) / 100)%10)),      \
  ('0' + (((n) / 10)%10)),       \
  ('0' +  ((n) % 10))
```

Definition at line 707 of file CMakeCXXCompilerId.cpp.

6.5.1.6 HEX

```
#define HEX(
              n )
```

Value:

```
  ('0' + ((n)>>28 & 0xF)), \
  ('0' + ((n)>>24 & 0xF)), \
  ('0' + ((n)>>20 & 0xF)), \
  ('0' + ((n)>>16 & 0xF)), \
  ('0' + ((n)>>12 & 0xF)), \
  ('0' + ((n)>>8  & 0xF)), \
  ('0' + ((n)>>4  & 0xF)), \
  ('0' + ((n)     & 0xF))
```

Definition at line 718 of file CMakeCXXCompilerId.cpp.

6.5.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```

Definition at line 545 of file CMakeCXXCompilerId.cpp.

6.5.1.8 STRINGIFY

```
#define STRINGIFY(
              X ) STRINGIFY_HELPER(X)
```

Definition at line 435 of file CMakeCXXCompilerId.cpp.

6.5.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(
              X ) #X
```

Definition at line 434 of file CMakeCXXCompilerId.cpp.

6.5.2 Function Documentation

6.5.2.1 main()

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

Definition at line 832 of file CMakeCXXCompilerId.cpp.

6.5.3 Variable Documentation

6.5.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

Definition at line 784 of file CMakeCXXCompilerId.cpp.

6.5.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

Definition at line 421 of file CMakeCXXCompilerId.cpp.

6.5.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["



  "OFF"

"]"
```

Definition at line 820 of file CMakeCXXCompilerId.cpp.

6.5.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
= "INFO" ":" "standard_default["









  "98"

"]"
```

Definition at line 804 of file CMakeCXXCompilerId.cpp.

6.5.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

Definition at line 783 of file CMakeCXXCompilerId.cpp.

6.6 CMakeCXXCompilerId.cpp

Go to the documentation of this file.

```
00001 /* This source file must have a .cpp extension so that all C++ compilers
00002    recognize the extension without flags.  Borland does not know .cxx for
00003    example.  */
00004 #ifndef __cplusplus
00005 # error "A C compiler has been selected for C++."
00006 #endif
00007
00008 #if !defined(__has_include)
00009 /* If the compiler does not have __has_include, pretend the answer is
00010    always no.  */
00011 #  define __has_include(x) 0
00012 #endif
00013
00014
00015 /* Version number components: V=Version, R=Revision, P=Patch
00016    Version date components:   YYYY=Year, MM=Month,   DD=Day  */
00017
00018 #if defined(__COMO__)
00019 # define COMPILER_ID "Comeau"
00020   /* __COMO_VERSION__ = VRR */
00021 # define COMPILER_VERSION_MAJOR DEC(__COMO_VERSION__ / 100)
00022 # define COMPILER_VERSION_MINOR DEC(__COMO_VERSION__ % 100)
00023
00024 #elif defined(__INTEL_COMPILER) || defined(__ICC)
00025 # define COMPILER_ID "Intel"
00026 # if defined(_MSC_VER)
00027 #  define SIMULATE_ID "MSVC"
00028 # endif
00029 # if defined(__GNUC__)
00030 #  define SIMULATE_ID "GNU"
00031 # endif
00032   /* __INTEL_COMPILER = VRP prior to 2021, and then VVVV for 2021 and later,
00033      except that a few beta releases use the old format with V=2021.  */
00034 # if __INTEL_COMPILER < 2021 || __INTEL_COMPILER == 202110 || __INTEL_COMPILER == 202111
00035 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER/100)
00036 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER/10 % 10)
00037 #  if defined(__INTEL_COMPILER_UPDATE)
00038 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER_UPDATE)
00039 #  else
00040 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER   % 10)
00041 #  endif
00042 # else
00043 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER)
00044 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER_UPDATE)
00045   /* The third version component from --version is an update index,
00046      but no macro is provided for it.  */
00047 #  define COMPILER_VERSION_PATCH DEC(0)
00048 # endif
00049 # if defined(__INTEL_COMPILER_BUILD_DATE)
00050   /* __INTEL_COMPILER_BUILD_DATE = YYYYMMDD */
00051 #  define COMPILER_VERSION_TWEAK DEC(__INTEL_COMPILER_BUILD_DATE)
00052 # endif
00053 # if defined(_MSC_VER)
00054   /* _MSC_VER = VVRR */
00055 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00056 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00057 # endif
00058 # if defined(__GNUC__)
00059 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00060 # elif defined(__GNUG__)
00061 #  define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00062 # endif
00063 # if defined(__GNUC_MINOR__)
00064 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00065 # endif
00066 # if defined(__GNUC_PATCHLEVEL__)
00067 #  define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00068 # endif
00069
```

```
00070 #elif (defined(__clang__) && defined(__INTEL_CLANG_COMPILER)) || defined(__INTEL_LLVM_COMPILER)
00071 # define COMPILER_ID "IntelLLVM"
00072 #if defined(_MSC_VER)
00073 # define SIMULATE_ID "MSVC"
00074 #endif
00075 #if defined(__GNUC__)
00076 # define SIMULATE_ID "GNU"
00077 #endif
00078 /* __INTEL_LLVM_COMPILER = VVVVRP prior to 2021.2.0, VVVVRRPP for 2021.2.0 and
00079  * later.  Look for 6 digit vs. 8 digit version number to decide encoding.
00080  * VVVV is no smaller than the current year when a version is released.
00081  */
00082 #if __INTEL_LLVM_COMPILER < 1000000L
00083 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/100)
00084 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/10 % 10)
00085 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER    % 10)
00086 #else
00087 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/10000)
00088 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/100 % 100)
00089 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER     % 100)
00090 #endif
00091 #if defined(_MSC_VER)
00092   /* _MSC_VER = VVRR */
00093 # define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00094 # define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00095 #endif
00096 #if defined(__GNUC__)
00097 # define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00098 #elif defined(__GNUG__)
00099 # define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00100 #endif
00101 #if defined(__GNUC_MINOR__)
00102 # define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00103 #endif
00104 #if defined(__GNUC_PATCHLEVEL__)
00105 # define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00106 #endif
00107
00108 #elif defined(__PATHCC__)
00109 # define COMPILER_ID "PathScale"
00110 # define COMPILER_VERSION_MAJOR DEC(__PATHCC__)
00111 # define COMPILER_VERSION_MINOR DEC(__PATHCC_MINOR__)
00112 # if defined(__PATHCC_PATCHLEVEL__)
00113 #  define COMPILER_VERSION_PATCH DEC(__PATHCC_PATCHLEVEL__)
00114 # endif
00115
00116 #elif defined(__BORLANDC__) && defined(__CODEGEARC_VERSION__)
00117 # define COMPILER_ID "Embarcadero"
00118 # define COMPILER_VERSION_MAJOR HEX(__CODEGEARC_VERSION__»24 & 0x00FF)
00119 # define COMPILER_VERSION_MINOR HEX(__CODEGEARC_VERSION__»16 & 0x00FF)
00120 # define COMPILER_VERSION_PATCH DEC(__CODEGEARC_VERSION__     & 0xFFFF)
00121
00122 #elif defined(__BORLANDC__)
00123 # define COMPILER_ID "Borland"
00124   /* __BORLANDC__ = 0xVRR */
00125 # define COMPILER_VERSION_MAJOR HEX(__BORLANDC__»8)
00126 # define COMPILER_VERSION_MINOR HEX(__BORLANDC__ & 0xFF)
00127
00128 #elif defined(__WATCOMC__) && __WATCOMC__ < 1200
00129 # define COMPILER_ID "Watcom"
00130    /* __WATCOMC__ = VVRR */
00131 # define COMPILER_VERSION_MAJOR DEC(__WATCOMC__ / 100)
00132 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00133 # if (__WATCOMC__ % 10) > 0
00134 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00135 # endif
00136
00137 #elif defined(__WATCOMC__)
00138 # define COMPILER_ID "OpenWatcom"
00139    /* __WATCOMC__ = VVRP + 1100 */
00140 # define COMPILER_VERSION_MAJOR DEC((__WATCOMC__ - 1100) / 100)
00141 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00142 # if (__WATCOMC__ % 10) > 0
00143 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00144 # endif
00145
00146 #elif defined(__SUNPRO_CC)
00147 # define COMPILER_ID "SunPro"
00148 # if __SUNPRO_CC >= 0x5100
00149    /* __SUNPRO_CC = 0xVRRP */
00150 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_CC»12)
00151 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_CC»4 & 0xFF)
00152 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_CC    & 0xF)
00153 # else
00154    /* __SUNPRO_CC = 0xVRP */
00155 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_CC»8)
00156 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_CC»4 & 0xF)
```

```
00157 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_CC    & 0xF)
00158 # endif
00159
00160 #elif defined(__HP_aCC)
00161 # define COMPILER_ID "HP"
00162   /* __HP_aCC = VVRRPP */
00163 # define COMPILER_VERSION_MAJOR DEC(__HP_aCC/10000)
00164 # define COMPILER_VERSION_MINOR DEC(__HP_aCC/100 % 100)
00165 # define COMPILER_VERSION_PATCH DEC(__HP_aCC     % 100)
00166
00167 #elif defined(__DECCXX)
00168 # define COMPILER_ID "Compaq"
00169   /* __DECCXX_VER = VVRRTPPPP */
00170 # define COMPILER_VERSION_MAJOR DEC(__DECCXX_VER/10000000)
00171 # define COMPILER_VERSION_MINOR DEC(__DECCXX_VER/100000  % 100)
00172 # define COMPILER_VERSION_PATCH DEC(__DECCXX_VER         % 10000)
00173
00174 #elif defined(__IBMCPP__) && defined(__COMPILER_VER__)
00175 # define COMPILER_ID "zOS"
00176   /* __IBMCPP__ = VRP */
00177 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00178 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00179 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00180
00181 #elif defined(__open_xl__) && defined(__clang__)
00182 # define COMPILER_ID "IBMClang"
00183 # define COMPILER_VERSION_MAJOR DEC(__open_xl_version__)
00184 # define COMPILER_VERSION_MINOR DEC(__open_xl_release__)
00185 # define COMPILER_VERSION_PATCH DEC(__open_xl_modification__)
00186 # define COMPILER_VERSION_TWEAK DEC(__open_xl_ptf_fix_level__)
00187
00188
00189 #elif defined(__ibmxl__) && defined(__clang__)
00190 # define COMPILER_ID "XLClang"
00191 # define COMPILER_VERSION_MAJOR DEC(__ibmxl_version__)
00192 # define COMPILER_VERSION_MINOR DEC(__ibmxl_release__)
00193 # define COMPILER_VERSION_PATCH DEC(__ibmxl_modification__)
00194 # define COMPILER_VERSION_TWEAK DEC(__ibmxl_ptf_fix_level__)
00195
00196
00197 #elif defined(__IBMCPP__) && !defined(__COMPILER_VER__) && __IBMCPP__ >= 800
00198 # define COMPILER_ID "XL"
00199   /* __IBMCPP__ = VRP */
00200 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00201 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00202 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00203
00204 #elif defined(__IBMCPP__) && !defined(__COMPILER_VER__) && __IBMCPP__ < 800
00205 # define COMPILER_ID "VisualAge"
00206   /* __IBMCPP__ = VRP */
00207 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00208 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00209 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00210
00211 #elif defined(__NVCOMPILER)
00212 # define COMPILER_ID "NVHPC"
00213 # define COMPILER_VERSION_MAJOR DEC(__NVCOMPILER_MAJOR__)
00214 # define COMPILER_VERSION_MINOR DEC(__NVCOMPILER_MINOR__)
00215 # if defined(__NVCOMPILER_PATCHLEVEL__)
00216 #  define COMPILER_VERSION_PATCH DEC(__NVCOMPILER_PATCHLEVEL__)
00217 # endif
00218
00219 #elif defined(__PGI)
00220 # define COMPILER_ID "PGI"
00221 # define COMPILER_VERSION_MAJOR DEC(__PGIC__)
00222 # define COMPILER_VERSION_MINOR DEC(__PGIC_MINOR__)
00223 # if defined(__PGIC_PATCHLEVEL__)
00224 #  define COMPILER_VERSION_PATCH DEC(__PGIC_PATCHLEVEL__)
00225 # endif
00226
00227 #elif defined(_CRAYC)
00228 # define COMPILER_ID "Cray"
00229 # define COMPILER_VERSION_MAJOR DEC(_RELEASE_MAJOR)
00230 # define COMPILER_VERSION_MINOR DEC(_RELEASE_MINOR)
00231
00232 #elif defined(__TI_COMPILER_VERSION__)
00233 # define COMPILER_ID "TI"
00234   /* __TI_COMPILER_VERSION__ = VVVRRRPPP */
00235 # define COMPILER_VERSION_MAJOR DEC(__TI_COMPILER_VERSION__/1000000)
00236 # define COMPILER_VERSION_MINOR DEC(__TI_COMPILER_VERSION__/1000   % 1000)
00237 # define COMPILER_VERSION_PATCH DEC(__TI_COMPILER_VERSION__        % 1000)
00238
00239 #elif defined(__CLANG_FUJITSU)
00240 # define COMPILER_ID "FujitsuClang"
00241 # define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00242 # define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00243 # define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
```

```
00244 # define COMPILER_VERSION_INTERNAL_STR __clang_version__
00245
00246
00247 #elif defined(__FUJITSU)
00248 # define COMPILER_ID "Fujitsu"
00249 # if defined(__FCC_version__)
00250 #   define COMPILER_VERSION __FCC_version__
00251 # elif defined(__FCC_major__)
00252 #   define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00253 #   define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00254 #   define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00255 # endif
00256 # if defined(__fcc_version)
00257 #   define COMPILER_VERSION_INTERNAL DEC(__fcc_version)
00258 # elif defined(__FCC_VERSION)
00259 #   define COMPILER_VERSION_INTERNAL DEC(__FCC_VERSION)
00260 # endif
00261
00262
00263 #elif defined(__ghs__)
00264 # define COMPILER_ID "GHS"
00265 /* __GHS_VERSION_NUMBER = VVVVRP */
00266 # ifdef __GHS_VERSION_NUMBER
00267 # define COMPILER_VERSION_MAJOR DEC(__GHS_VERSION_NUMBER / 100)
00268 # define COMPILER_VERSION_MINOR DEC(__GHS_VERSION_NUMBER / 10 % 10)
00269 # define COMPILER_VERSION_PATCH DEC(__GHS_VERSION_NUMBER      % 10)
00270 # endif
00271
00272 #elif defined(__TASKING__)
00273 # define COMPILER_ID "Tasking"
00274   # define COMPILER_VERSION_MAJOR DEC(__VERSION__/1000)
00275   # define COMPILER_VERSION_MINOR DEC(__VERSION__ % 100)
00276 # define COMPILER_VERSION_INTERNAL DEC(__VERSION__)
00277
00278 #elif defined(__SCO_VERSION__)
00279 # define COMPILER_ID "SCO"
00280
00281 #elif defined(__ARMCC_VERSION) && !defined(__clang__)
00282 # define COMPILER_ID "ARMCC"
00283 #if __ARMCC_VERSION >= 1000000
00284   /* __ARMCC_VERSION = VRRPPPP */
00285   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/1000000)
00286   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 100)
00287   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION     % 10000)
00288 #else
00289   /* __ARMCC_VERSION = VRPPPP */
00290   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/100000)
00291   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 10)
00292   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION    % 10000)
00293 #endif
00294
00295
00296 #elif defined(__clang__) && defined(__apple_build_version__)
00297 # define COMPILER_ID "AppleClang"
00298 # if defined(_MSC_VER)
00299 #  define SIMULATE_ID "MSVC"
00300 # endif
00301 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00302 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00303 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00304 # if defined(_MSC_VER)
00305   /* _MSC_VER = VVRR */
00306 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00307 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00308 # endif
00309 # define COMPILER_VERSION_TWEAK DEC(__apple_build_version__)
00310
00311 #elif defined(__clang__) && defined(__ARMCOMPILER_VERSION)
00312 # define COMPILER_ID "ARMClang"
00313   # define COMPILER_VERSION_MAJOR DEC(__ARMCOMPILER_VERSION/1000000)
00314   # define COMPILER_VERSION_MINOR DEC(__ARMCOMPILER_VERSION/10000 % 100)
00315   # define COMPILER_VERSION_PATCH DEC(__ARMCOMPILER_VERSION     % 10000)
00316 # define COMPILER_VERSION_INTERNAL DEC(__ARMCOMPILER_VERSION)
00317
00318 #elif defined(__clang__)
00319 # define COMPILER_ID "Clang"
00320 # if defined(_MSC_VER)
00321 #  define SIMULATE_ID "MSVC"
00322 # endif
00323 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00324 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00325 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00326 # if defined(_MSC_VER)
00327   /* _MSC_VER = VVRR */
00328 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00329 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00330 # endif
```

```
00331
00332 #elif defined(__LCC__) && (defined(__GNUC__) || defined(__GNUG__) || defined(__MCST__))
00333 # define COMPILER_ID "LCC"
00334 # define COMPILER_VERSION_MAJOR DEC(1)
00335 # if defined(__LCC__)
00336 #  define COMPILER_VERSION_MINOR DEC(__LCC__- 100)
00337 # endif
00338 # if defined(__LCC_MINOR__)
00339 #  define COMPILER_VERSION_PATCH DEC(__LCC_MINOR__)
00340 # endif
00341 # if defined(__GNUC__) && defined(__GNUC_MINOR__)
00342 #  define SIMULATE_ID "GNU"
00343 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00344 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00345 #  if defined(__GNUC_PATCHLEVEL__)
00346 #   define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00347 #  endif
00348 # endif
00349
00350 #elif defined(__GNUC__) || defined(__GNUG__)
00351 # define COMPILER_ID "GNU"
00352 # if defined(__GNUC__)
00353 #  define COMPILER_VERSION_MAJOR DEC(__GNUC__)
00354 # else
00355 #  define COMPILER_VERSION_MAJOR DEC(__GNUG__)
00356 # endif
00357 # if defined(__GNUC_MINOR__)
00358 #  define COMPILER_VERSION_MINOR DEC(__GNUC_MINOR__)
00359 # endif
00360 # if defined(__GNUC_PATCHLEVEL__)
00361 #  define COMPILER_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00362 # endif
00363
00364 #elif defined(_MSC_VER)
00365 # define COMPILER_ID "MSVC"
00366   /* _MSC_VER = VVRR */
00367 # define COMPILER_VERSION_MAJOR DEC(_MSC_VER / 100)
00368 # define COMPILER_VERSION_MINOR DEC(_MSC_VER % 100)
00369 # if defined(_MSC_FULL_VER)
00370 #  if _MSC_VER >= 1400
00371     /* _MSC_FULL_VER = VVRRPPPPP */
00372 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 100000)
00373 #  else
00374     /* _MSC_FULL_VER = VVRRPPPP */
00375 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 10000)
00376 #  endif
00377 # endif
00378 # if defined(_MSC_BUILD)
00379 #  define COMPILER_VERSION_TWEAK DEC(_MSC_BUILD)
00380 # endif
00381
00382 #elif defined(_ADI_COMPILER)
00383 # define COMPILER_ID "ADSP"
00384 #if defined(__VERSIONNUM__)
00385   /* __VERSIONNUM__ = 0xVVRRPTT */
00386 #  define COMPILER_VERSION_MAJOR DEC(__VERSIONNUM__ >> 24 & 0xFF)
00387 #  define COMPILER_VERSION_MINOR DEC(__VERSIONNUM__ >> 16 & 0xFF)
00388 #  define COMPILER_VERSION_PATCH DEC(__VERSIONNUM__ >> 8 & 0xFF)
00389 #  define COMPILER_VERSION_TWEAK DEC(__VERSIONNUM__ & 0xFF)
00390 #endif
00391
00392 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00393 # define COMPILER_ID "IAR"
00394 # if defined(__VER__) && defined(__ICCARM__)
00395 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 1000000)
00396 #  define COMPILER_VERSION_MINOR DEC(((__VER__) / 1000) % 1000)
00397 #  define COMPILER_VERSION_PATCH DEC((__VER__) % 1000)
00398 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00399 # elif defined(__VER__) && (defined(__ICCAVR__) || defined(__ICCRX__) || defined(__ICCRH850__) ||
      defined(__ICCRL78__) || defined(__ICC430__) || defined(__ICCRISCV__) || defined(__ICCV850__) ||
      defined(__ICC8051__) || defined(__ICCSTM8__))
00400 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 100)
00401 #  define COMPILER_VERSION_MINOR DEC((__VER__) - (((__VER__) / 100)*100))
00402 #  define COMPILER_VERSION_PATCH DEC(__SUBVERSION__)
00403 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00404 # endif
00405
00406
00407 /* These compilers are either not known or too old to define an
00408   identification macro.  Try to identify the platform and guess that
00409   it is the native compiler.  */
00410 #elif defined(__hpux) || defined(__hpua)
00411 # define COMPILER_ID "HP"
00412
00413 #else /* unknown compiler */
00414 # define COMPILER_ID ""
00415 #endif
```

```
00416
00417 /* Construct the string literal in pieces to prevent the source from
00418    getting matched.  Store it in a pointer rather than an array
00419    because some compilers will just produce instructions to fill the
00420    array rather than assigning a pointer to a static array.  */
00421 char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]";
00422 #ifdef SIMULATE_ID
00423 char const* info_simulate = "INFO" ":" "simulate[" SIMULATE_ID "]";
00424 #endif
00425
00426 #ifdef __QNXNTO__
00427 char const* qnxnto = "INFO" ":" "qnxnto[]";
00428 #endif
00429
00430 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00431 char const *info_cray = "INFO" ":" "compiler_wrapper[CrayPrgEnv]";
00432 #endif
00433
00434 #define STRINGIFY_HELPER(X) #X
00435 #define STRINGIFY(X) STRINGIFY_HELPER(X)
00436
00437 /* Identify known platforms by name.  */
00438 #if defined(__linux) || defined(__linux__) || defined(linux)
00439 # define PLATFORM_ID "Linux"
00440
00441 #elif defined(__MSYS__)
00442 # define PLATFORM_ID "MSYS"
00443
00444 #elif defined(__CYGWIN__)
00445 # define PLATFORM_ID "Cygwin"
00446
00447 #elif defined(__MINGW32__)
00448 # define PLATFORM_ID "MinGW"
00449
00450 #elif defined(__APPLE__)
00451 # define PLATFORM_ID "Darwin"
00452
00453 #elif defined(_WIN32) || defined(__WIN32__) || defined(WIN32)
00454 # define PLATFORM_ID "Windows"
00455
00456 #elif defined(__FreeBSD__) || defined(__FreeBSD)
00457 # define PLATFORM_ID "FreeBSD"
00458
00459 #elif defined(__NetBSD__) || defined(__NetBSD)
00460 # define PLATFORM_ID "NetBSD"
00461
00462 #elif defined(__OpenBSD__) || defined(__OPENBSD)
00463 # define PLATFORM_ID "OpenBSD"
00464
00465 #elif defined(__sun) || defined(sun)
00466 # define PLATFORM_ID "SunOS"
00467
00468 #elif defined(_AIX) || defined(__AIX) || defined(__AIX__) || defined(__aix) || defined(__aix__)
00469 # define PLATFORM_ID "AIX"
00470
00471 #elif defined(__hpux) || defined(__hpux__)
00472 # define PLATFORM_ID "HP-UX"
00473
00474 #elif defined(__HAIKU__)
00475 # define PLATFORM_ID "Haiku"
00476
00477 #elif defined(__BeOS) || defined(__BEOS__) || defined(_BEOS)
00478 # define PLATFORM_ID "BeOS"
00479
00480 #elif defined(__QNX__) || defined(__QNXNTO__)
00481 # define PLATFORM_ID "QNX"
00482
00483 #elif defined(__tru64) || defined(_tru64) || defined(__TRU64__)
00484 # define PLATFORM_ID "Tru64"
00485
00486 #elif defined(__riscos) || defined(__riscos__)
00487 # define PLATFORM_ID "RISCos"
00488
00489 #elif defined(__sinix) || defined(__sinix__) || defined(__SINIX__)
00490 # define PLATFORM_ID "SINIX"
00491
00492 #elif defined(__UNIX_SV__)
00493 # define PLATFORM_ID "UNIX_SV"
00494
00495 #elif defined(__bsdos__)
00496 # define PLATFORM_ID "BSDOS"
00497
00498 #elif defined(_MPRAS) || defined(MPRAS)
00499 # define PLATFORM_ID "MP-RAS"
00500
00501 #elif defined(__osf) || defined(__osf__)
00502 # define PLATFORM_ID "OSF1"
```

```
00503
00504 #elif defined(_SCO_SV) || defined(SCO_SV) || defined(sco_sv)
00505 # define PLATFORM_ID "SCO_SV"
00506
00507 #elif defined(__ultrix) || defined(__ultrix__) || defined(_ULTRIX)
00508 # define PLATFORM_ID "ULTRIX"
00509
00510 #elif defined(__XENIX__) || defined(_XENIX) || defined(XENIX)
00511 # define PLATFORM_ID "Xenix"
00512
00513 #elif defined(__WATCOMC__)
00514 # if defined(__LINUX__)
00515 #  define PLATFORM_ID "Linux"
00516
00517 # elif defined(__DOS__)
00518 #  define PLATFORM_ID "DOS"
00519
00520 # elif defined(__OS2__)
00521 #  define PLATFORM_ID "OS2"
00522
00523 # elif defined(__WINDOWS__)
00524 #  define PLATFORM_ID "Windows3x"
00525
00526 # elif defined(__VXWORKS__)
00527 #  define PLATFORM_ID "VxWorks"
00528
00529 # else /* unknown platform */
00530 #  define PLATFORM_ID
00531 # endif
00532
00533 #elif defined(__INTEGRITY)
00534 # if defined(INT_178B)
00535 #  define PLATFORM_ID "Integrity178"
00536
00537 # else /* regular Integrity */
00538 #  define PLATFORM_ID "Integrity"
00539 # endif
00540
00541 # elif defined(_ADI_COMPILER)
00542 #  define PLATFORM_ID "ADSP"
00543
00544 #else /* unknown platform */
00545 # define PLATFORM_ID
00546
00547 #endif
00548
00549 /* For windows compilers MSVC and Intel we can determine
00550    the architecture of the compiler being used.  This is because
00551    the compilers do not have flags that can change the architecture,
00552    but rather depend on which compiler is being used
00553 */
00554 #if defined(_WIN32) && defined(_MSC_VER)
00555 # if defined(_M_IA64)
00556 #  define ARCHITECTURE_ID "IA64"
00557
00558 # elif defined(_M_ARM64EC)
00559 #  define ARCHITECTURE_ID "ARM64EC"
00560
00561 # elif defined(_M_X64) || defined(_M_AMD64)
00562 #  define ARCHITECTURE_ID "x64"
00563
00564 # elif defined(_M_IX86)
00565 #  define ARCHITECTURE_ID "X86"
00566
00567 # elif defined(_M_ARM64)
00568 #  define ARCHITECTURE_ID "ARM64"
00569
00570 # elif defined(_M_ARM)
00571 #  if _M_ARM == 4
00572 #   define ARCHITECTURE_ID "ARMV4I"
00573 #  elif _M_ARM == 5
00574 #   define ARCHITECTURE_ID "ARMV5I"
00575 #  else
00576 #   define ARCHITECTURE_ID "ARMV" STRINGIFY(_M_ARM)
00577 #  endif
00578
00579 # elif defined(_M_MIPS)
00580 #  define ARCHITECTURE_ID "MIPS"
00581
00582 # elif defined(_M_SH)
00583 #  define ARCHITECTURE_ID "SHx"
00584
00585 # else /* unknown architecture */
00586 #  define ARCHITECTURE_ID ""
00587 # endif
00588
00589 #elif defined(__WATCOMC__)
```

```
00590 # if defined(_M_I86)
00591 #  define ARCHITECTURE_ID "I86"
00592
00593 # elif defined(_M_IX86)
00594 #  define ARCHITECTURE_ID "X86"
00595
00596 # else /* unknown architecture */
00597 #  define ARCHITECTURE_ID ""
00598 # endif
00599
00600 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00601 # if defined(__ICCARM__)
00602 #  define ARCHITECTURE_ID "ARM"
00603
00604 # elif defined(__ICCRX__)
00605 #  define ARCHITECTURE_ID "RX"
00606
00607 # elif defined(__ICCRH850__)
00608 #  define ARCHITECTURE_ID "RH850"
00609
00610 # elif defined(__ICCRL78__)
00611 #  define ARCHITECTURE_ID "RL78"
00612
00613 # elif defined(__ICCRISCV__)
00614 #  define ARCHITECTURE_ID "RISCV"
00615
00616 # elif defined(__ICCAVR__)
00617 #  define ARCHITECTURE_ID "AVR"
00618
00619 # elif defined(__ICC430__)
00620 #  define ARCHITECTURE_ID "MSP430"
00621
00622 # elif defined(__ICCV850__)
00623 #  define ARCHITECTURE_ID "V850"
00624
00625 # elif defined(__ICC8051__)
00626 #  define ARCHITECTURE_ID "8051"
00627
00628 # elif defined(__ICCSTM8__)
00629 #  define ARCHITECTURE_ID "STM8"
00630
00631 # else /* unknown architecture */
00632 #  define ARCHITECTURE_ID ""
00633 # endif
00634
00635 #elif defined(__ghs__)
00636 # if defined(__PPC64__)
00637 #  define ARCHITECTURE_ID "PPC64"
00638
00639 # elif defined(__ppc__)
00640 #  define ARCHITECTURE_ID "PPC"
00641
00642 # elif defined(__ARM__)
00643 #  define ARCHITECTURE_ID "ARM"
00644
00645 # elif defined(__x86_64__)
00646 #  define ARCHITECTURE_ID "x64"
00647
00648 # elif defined(__i386__)
00649 #  define ARCHITECTURE_ID "X86"
00650
00651 # else /* unknown architecture */
00652 #  define ARCHITECTURE_ID ""
00653 # endif
00654
00655 #elif defined(__TI_COMPILER_VERSION__)
00656 # if defined(__TI_ARM__)
00657 #  define ARCHITECTURE_ID "ARM"
00658
00659 # elif defined(__MSP430__)
00660 #  define ARCHITECTURE_ID "MSP430"
00661
00662 # elif defined(__TMS320C28XX__)
00663 #  define ARCHITECTURE_ID "TMS320C28x"
00664
00665 # elif defined(__TMS320C6X__) || defined(_TMS320C6X)
00666 #  define ARCHITECTURE_ID "TMS320C6x"
00667
00668 # else /* unknown architecture */
00669 #  define ARCHITECTURE_ID ""
00670 # endif
00671
00672 # elif defined(__ADSPSHARC__)
00673 #  define ARCHITECTURE_ID "SHARC"
00674
00675 # elif defined(__ADSPBLACKFIN__)
00676 #  define ARCHITECTURE_ID "Blackfin"
```

```
00677
00678 #elif defined(__TASKING__)
00679
00680 # if defined(__CTC__) || defined(__CPTC__)
00681 #  define ARCHITECTURE_ID "TriCore"
00682
00683 # elif defined(__CMCS__)
00684 #  define ARCHITECTURE_ID "MCS"
00685
00686 # elif defined(__CARM__)
00687 #  define ARCHITECTURE_ID "ARM"
00688
00689 # elif defined(__CARC__)
00690 #  define ARCHITECTURE_ID "ARC"
00691
00692 # elif defined(__C51__)
00693 #  define ARCHITECTURE_ID "8051"
00694
00695 # elif defined(__CPCP__)
00696 #  define ARCHITECTURE_ID "PCP"
00697
00698 # else
00699 #  define ARCHITECTURE_ID ""
00700 # endif
00701
00702 #else
00703 #  define ARCHITECTURE_ID
00704 #endif
00705
00706 /* Convert integer to decimal digit literals.  */
00707 #define DEC(n)                   \
00708   ('0' + (((n) / 10000000)%10)), \
00709   ('0' + (((n) / 1000000)%10)),  \
00710   ('0' + (((n) / 100000)%10)),   \
00711   ('0' + (((n) / 10000)%10)),    \
00712   ('0' + (((n) / 1000)%10)),     \
00713   ('0' + (((n) / 100)%10)),      \
00714   ('0' + (((n) / 10)%10)),       \
00715   ('0' +  ((n) % 10))
00716
00717 /* Convert integer to hex digit literals.  */
00718 #define HEX(n)             \
00719   ('0' + ((n)>>28 & 0xF)), \
00720   ('0' + ((n)>>24 & 0xF)), \
00721   ('0' + ((n)>>20 & 0xF)), \
00722   ('0' + ((n)>>16 & 0xF)), \
00723   ('0' + ((n)>>12 & 0xF)), \
00724   ('0' + ((n)>>8  & 0xF)), \
00725   ('0' + ((n)>>4  & 0xF)), \
00726   ('0' + ((n)     & 0xF))
00727
00728 /* Construct a string literal encoding the version number. */
00729 #ifdef COMPILER_VERSION
00730 char const* info_version = "INFO" ":" "compiler_version[" COMPILER_VERSION "]";
00731
00732 /* Construct a string literal encoding the version number components. */
00733 #elif defined(COMPILER_VERSION_MAJOR)
00734 char const info_version[] = {
00735   'I', 'N', 'F', 'O', ':',
00736   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','[',
00737   COMPILER_VERSION_MAJOR,
00738 # ifdef COMPILER_VERSION_MINOR
00739   '.', COMPILER_VERSION_MINOR,
00740 #  ifdef COMPILER_VERSION_PATCH
00741    '.', COMPILER_VERSION_PATCH,
00742 #   ifdef COMPILER_VERSION_TWEAK
00743     '.', COMPILER_VERSION_TWEAK,
00744 #   endif
00745 #  endif
00746 # endif
00747   ']','\0'};
00748 #endif
00749
00750 /* Construct a string literal encoding the internal version number. */
00751 #ifdef COMPILER_VERSION_INTERNAL
00752 char const info_version_internal[] = {
00753   'I', 'N', 'F', 'O', ':',
00754   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','_',
00755   'i','n','t','e','r','n','a','l','[',
00756   COMPILER_VERSION_INTERNAL,']','\0'};
00757 #elif defined(COMPILER_VERSION_INTERNAL_STR)
00758 char const* info_version_internal = "INFO" ":" "compiler_version_internal["
      COMPILER_VERSION_INTERNAL_STR "]";
00759 #endif
00760
00761 /* Construct a string literal encoding the version number components. */
00762 #ifdef SIMULATE_VERSION_MAJOR
```

```
00763 char const info_simulate_version[] = {
00764   'I', 'N', 'F', 'O', ':',
00765   's','i','m','u','l','a','t','e','_','v','e','r','s','i','o','n','[',
00766   SIMULATE_VERSION_MAJOR,
00767 # ifdef SIMULATE_VERSION_MINOR
00768   '.', SIMULATE_VERSION_MINOR,
00769 #  ifdef SIMULATE_VERSION_PATCH
00770   '.', SIMULATE_VERSION_PATCH,
00771 #   ifdef SIMULATE_VERSION_TWEAK
00772     '.', SIMULATE_VERSION_TWEAK,
00773 #   endif
00774 #  endif
00775 # endif
00776   ']','\0'};
00777 #endif
00778
00779 /* Construct the string literal in pieces to prevent the source from
00780    getting matched.  Store it in a pointer rather than an array
00781    because some compilers will just produce instructions to fill the
00782    array rather than assigning a pointer to a static array.  */
00783 char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]";
00784 char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]";
00785
00786
00787
00788 #if defined(__INTEL_COMPILER) && defined(_MSVC_LANG) && _MSVC_LANG < 201403L
00789 #  if defined(__INTEL_CXX11_MODE__)
00790 #    if defined(__cpp_aggregate_nsdmi)
00791 #      define CXX_STD 201402L
00792 #    else
00793 #      define CXX_STD 201103L
00794 #    endif
00795 #  else
00796 #    define CXX_STD 199711L
00797 #  endif
00798 #elif defined(_MSC_VER) && defined(_MSVC_LANG)
00799 #  define CXX_STD _MSVC_LANG
00800 #else
00801 #  define CXX_STD __cplusplus
00802 #endif
00803
00804 const char* info_language_standard_default = "INFO" ":" "standard_default["
00805 #if CXX_STD > 202002L
00806   "23"
00807 #elif CXX_STD > 201703L
00808   "20"
00809 #elif CXX_STD >= 201703L
00810   "17"
00811 #elif CXX_STD >= 201402L
00812   "14"
00813 #elif CXX_STD >= 201103L
00814   "11"
00815 #else
00816   "98"
00817 #endif
00818 "]";
00819
00820 const char* info_language_extensions_default = "INFO" ":" "extensions_default["
00821 #if (defined(__clang__) || defined(__GNUC__) || defined(__xlC__) ||           \
00822      defined(__TI_COMPILER_VERSION__)) &&                                     \
00823   !defined(__STRICT_ANSI__)
00824   "ON"
00825 #else
00826   "OFF"
00827 #endif
00828 "]";
00829
00830 /*--------------------------------------------------------------------------*/
00831
00832 int main(int argc, char* argv[])
00833 {
00834   int require = 0;
00835   require += info_compiler[argc];
00836   require += info_platform[argc];
00837   require += info_arch[argc];
00838 #ifdef COMPILER_VERSION_MAJOR
00839   require += info_version[argc];
00840 #endif
00841 #ifdef COMPILER_VERSION_INTERNAL
00842   require += info_version_internal[argc];
00843 #endif
00844 #ifdef SIMULATE_ID
00845   require += info_simulate[argc];
00846 #endif
00847 #ifdef SIMULATE_VERSION_MAJOR
00848   require += info_simulate_version[argc];
00849 #endif
```

```
00850 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00851   require += info_cray[argc];
00852 #endif
00853   require += info_language_standard_default[argc];
00854   require += info_language_extensions_default[argc];
00855   (void)argv;
00856   return require;
00857 }
```

6.7 build/default/CMakeFiles/3.25.1/CompilerIdCXX/CMake↩ CXXCompilerId.cpp File Reference

Macros

- #define __has_include(x) 0
- #define COMPILER_ID ""
- #define STRINGIFY_HELPER(X) #X
- #define STRINGIFY(X) STRINGIFY_HELPER(X)
- #define PLATFORM_ID
- #define ARCHITECTURE_ID
- #define DEC(n)
- #define HEX(n)
- #define CXX_STD __cplusplus

Functions

- int main (int argc, char ∗argv[ ])

Variables

- char const ∗ info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
- char const ∗ info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
- char const ∗ info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
- const char ∗ info_language_standard_default
- const char ∗ info_language_extensions_default

6.7.1 Macro Definition Documentation

6.7.1.1 __has_include

```
#define __has_include(
              x ) 0
```

Definition at line 11 of file CMakeCXXCompilerId.cpp.

6.7.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

Definition at line 703 of file CMakeCXXCompilerId.cpp.

6.7.1.3 COMPILER_ID

```
#define COMPILER_ID ""
```

Definition at line 414 of file CMakeCXXCompilerId.cpp.

6.7.1.4 CXX_STD

```
#define CXX_STD __cplusplus
```

Definition at line 801 of file CMakeCXXCompilerId.cpp.

6.7.1.5 DEC

```
#define DEC(
              n )
```

Value:

```
  ('0' + (((n) / 10000000)%10)), \
  ('0' + (((n) / 1000000)%10)),  \
  ('0' + (((n) / 100000)%10)),   \
  ('0' + (((n) / 10000)%10)),    \
  ('0' + (((n) / 1000)%10)),     \
  ('0' + (((n) / 100)%10)),      \
  ('0' + (((n) / 10)%10)),       \
  ('0' +  ((n) % 10))
```

Definition at line 707 of file CMakeCXXCompilerId.cpp.

6.7.1.6 HEX

```
#define HEX(
              n )
```

Value:

```
  ('0' + ((n)>>28 & 0xF)), \
  ('0' + ((n)>>24 & 0xF)), \
  ('0' + ((n)>>20 & 0xF)), \
  ('0' + ((n)>>16 & 0xF)), \
  ('0' + ((n)>>12 & 0xF)), \
  ('0' + ((n)>>8  & 0xF)), \
  ('0' + ((n)>>4  & 0xF)), \
  ('0' + ((n)     & 0xF))
```

Definition at line 718 of file CMakeCXXCompilerId.cpp.

6.7.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```

Definition at line 545 of file CMakeCXXCompilerId.cpp.

6.7.1.8 STRINGIFY

```
#define STRINGIFY(
              X ) STRINGIFY_HELPER(X)
```

Definition at line 435 of file CMakeCXXCompilerId.cpp.

6.7.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(
              X ) #X
```

Definition at line 434 of file CMakeCXXCompilerId.cpp.

6.7.2 Function Documentation

6.7.2.1 main()

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

Definition at line 832 of file CMakeCXXCompilerId.cpp.

6.7.3 Variable Documentation

6.7.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

Definition at line 784 of file CMakeCXXCompilerId.cpp.

6.7.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

Definition at line 421 of file CMakeCXXCompilerId.cpp.

6.7.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["



  "OFF"

"]"
```

Definition at line 820 of file CMakeCXXCompilerId.cpp.

6.7.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
= "INFO" ":" "standard_default["
```

```
  "98"
```

```
"]"
```

Definition at line 804 of file CMakeCXXCompilerId.cpp.

6.7.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

Definition at line 783 of file CMakeCXXCompilerId.cpp.

6.8 CMakeCXXCompilerId.cpp

Go to the documentation of this file.

```
00001 /* This source file must have a .cpp extension so that all C++ compilers
00002    recognize the extension without flags.  Borland does not know .cxx for
00003    example.  */
00004 #ifndef __cplusplus
00005 # error "A C compiler has been selected for C++."
00006 #endif
00007
00008 #if !defined(__has_include)
00009 /* If the compiler does not have __has_include, pretend the answer is
00010    always no.  */
00011 #  define __has_include(x) 0
00012 #endif
00013
00014
00015 /* Version number components: V=Version, R=Revision, P=Patch
00016    Version date components:   YYYY=Year, MM=Month,   DD=Day  */
00017
00018 #if defined(__COMO__)
00019 # define COMPILER_ID "Comeau"
00020   /* __COMO_VERSION__ = VRR */
00021 # define COMPILER_VERSION_MAJOR DEC(__COMO_VERSION__ / 100)
00022 # define COMPILER_VERSION_MINOR DEC(__COMO_VERSION__ % 100)
00023
00024 #elif defined(__INTEL_COMPILER) || defined(__ICC)
00025 # define COMPILER_ID "Intel"
00026 # if defined(_MSC_VER)
00027 #  define SIMULATE_ID "MSVC"
00028 # endif
00029 # if defined(__GNUC__)
00030 #  define SIMULATE_ID "GNU"
00031 # endif
00032   /* __INTEL_COMPILER = VRP prior to 2021, and then VVVV for 2021 and later,
00033      except that a few beta releases use the old format with V=2021.  */
00034 # if __INTEL_COMPILER < 2021 || __INTEL_COMPILER == 202110 || __INTEL_COMPILER == 202111
00035 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER/100)
00036 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER/10 % 10)
00037 #  if defined(__INTEL_COMPILER_UPDATE)
00038 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER_UPDATE)
00039 #  else
00040 #   define COMPILER_VERSION_PATCH DEC(__INTEL_COMPILER   % 10)
```

```
00041 #  endif
00042 # else
00043 #  define COMPILER_VERSION_MAJOR DEC(__INTEL_COMPILER)
00044 #  define COMPILER_VERSION_MINOR DEC(__INTEL_COMPILER_UPDATE)
00045    /* The third version component from --version is an update index,
00046       but no macro is provided for it.  */
00047 #  define COMPILER_VERSION_PATCH DEC(0)
00048 # endif
00049 # if defined(__INTEL_COMPILER_BUILD_DATE)
00050    /* __INTEL_COMPILER_BUILD_DATE = YYYYMMDD */
00051 #  define COMPILER_VERSION_TWEAK DEC(__INTEL_COMPILER_BUILD_DATE)
00052 # endif
00053 # if defined(_MSC_VER)
00054    /* _MSC_VER = VVRR */
00055 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00056 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00057 # endif
00058 # if defined(__GNUC__)
00059 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00060 # elif defined(__GNUG__)
00061 #  define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00062 # endif
00063 # if defined(__GNUC_MINOR__)
00064 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00065 # endif
00066 # if defined(__GNUC_PATCHLEVEL__)
00067 #  define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00068 # endif
00069
00070 #elif (defined(__clang__) && defined(__INTEL_CLANG_COMPILER)) || defined(__INTEL_LLVM_COMPILER)
00071 # define COMPILER_ID "IntelLLVM"
00072 #if defined(_MSC_VER)
00073 # define SIMULATE_ID "MSVC"
00074 #endif
00075 #if defined(__GNUC__)
00076 # define SIMULATE_ID "GNU"
00077 #endif
00078 /* __INTEL_LLVM_COMPILER = VVVVRP prior to 2021.2.0, VVVVRRPP for 2021.2.0 and
00079  * later.  Look for 6 digit vs. 8 digit version number to decide encoding.
00080  * VVVV is no smaller than the current year when a version is released.
00081  */
00082 #if __INTEL_LLVM_COMPILER < 1000000L
00083 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/100)
00084 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/10 % 10)
00085 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER    % 10)
00086 #else
00087 # define COMPILER_VERSION_MAJOR DEC(__INTEL_LLVM_COMPILER/10000)
00088 # define COMPILER_VERSION_MINOR DEC(__INTEL_LLVM_COMPILER/100 % 100)
00089 # define COMPILER_VERSION_PATCH DEC(__INTEL_LLVM_COMPILER     % 100)
00090 #endif
00091 #if defined(_MSC_VER)
00092   /* _MSC_VER = VVRR */
00093 # define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00094 # define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00095 #endif
00096 #if defined(__GNUC__)
00097 # define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00098 #elif defined(__GNUG__)
00099 # define SIMULATE_VERSION_MAJOR DEC(__GNUG__)
00100 #endif
00101 #if defined(__GNUC_MINOR__)
00102 # define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00103 #endif
00104 #if defined(__GNUC_PATCHLEVEL__)
00105 # define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00106 #endif
00107
00108 #elif defined(__PATHCC__)
00109 # define COMPILER_ID "PathScale"
00110 # define COMPILER_VERSION_MAJOR DEC(__PATHCC__)
00111 # define COMPILER_VERSION_MINOR DEC(__PATHCC_MINOR__)
00112 # if defined(__PATHCC_PATCHLEVEL__)
00113 #  define COMPILER_VERSION_PATCH DEC(__PATHCC_PATCHLEVEL__)
00114 # endif
00115
00116 #elif defined(__BORLANDC__) && defined(__CODEGEARC_VERSION__)
00117 # define COMPILER_ID "Embarcadero"
00118 # define COMPILER_VERSION_MAJOR HEX(__CODEGEARC_VERSION__»24 & 0x00FF)
00119 # define COMPILER_VERSION_MINOR HEX(__CODEGEARC_VERSION__»16 & 0x00FF)
00120 # define COMPILER_VERSION_PATCH DEC(__CODEGEARC_VERSION__     & 0xFFFF)
00121
00122 #elif defined(__BORLANDC__)
00123 # define COMPILER_ID "Borland"
00124   /* __BORLANDC__ = 0xVRR */
00125 # define COMPILER_VERSION_MAJOR HEX(__BORLANDC__»8)
00126 # define COMPILER_VERSION_MINOR HEX(__BORLANDC__ & 0xFF)
00127
```

```
00128 #elif defined(__WATCOMC__) && __WATCOMC__ < 1200
00129 # define COMPILER_ID "Watcom"
00130   /* __WATCOMC__ = VVRR */
00131 # define COMPILER_VERSION_MAJOR DEC(__WATCOMC__ / 100)
00132 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00133 # if (__WATCOMC__ % 10) > 0
00134 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00135 # endif
00136
00137 #elif defined(__WATCOMC__)
00138 # define COMPILER_ID "OpenWatcom"
00139   /* __WATCOMC__ = VVRP + 1100 */
00140 # define COMPILER_VERSION_MAJOR DEC((__WATCOMC__ - 1100) / 100)
00141 # define COMPILER_VERSION_MINOR DEC((__WATCOMC__ / 10) % 10)
00142 # if (__WATCOMC__ % 10) > 0
00143 #  define COMPILER_VERSION_PATCH DEC(__WATCOMC__ % 10)
00144 # endif
00145
00146 #elif defined(__SUNPRO_CC)
00147 # define COMPILER_ID "SunPro"
00148 # if __SUNPRO_CC >= 0x5100
00149   /* __SUNPRO_CC = 0xVRRP */
00150 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_CC»12)
00151 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_CC»4 & 0xFF)
00152 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_CC    & 0xF)
00153 # else
00154   /* __SUNPRO_CC = 0xVRP */
00155 #  define COMPILER_VERSION_MAJOR HEX(__SUNPRO_CC»8)
00156 #  define COMPILER_VERSION_MINOR HEX(__SUNPRO_CC»4 & 0xF)
00157 #  define COMPILER_VERSION_PATCH HEX(__SUNPRO_CC    & 0xF)
00158 # endif
00159
00160 #elif defined(__HP_aCC)
00161 # define COMPILER_ID "HP"
00162   /* __HP_aCC = VVRRPP */
00163 # define COMPILER_VERSION_MAJOR DEC(__HP_aCC/10000)
00164 # define COMPILER_VERSION_MINOR DEC(__HP_aCC/100 % 100)
00165 # define COMPILER_VERSION_PATCH DEC(__HP_aCC     % 100)
00166
00167 #elif defined(__DECCXX)
00168 # define COMPILER_ID "Compaq"
00169   /* __DECCXX_VER = VVRRTPPPP */
00170 # define COMPILER_VERSION_MAJOR DEC(__DECCXX_VER/10000000)
00171 # define COMPILER_VERSION_MINOR DEC(__DECCXX_VER/100000  % 100)
00172 # define COMPILER_VERSION_PATCH DEC(__DECCXX_VER         % 10000)
00173
00174 #elif defined(__IBMCPP__) && defined(__COMPILER_VER__)
00175 # define COMPILER_ID "zOS"
00176   /* __IBMCPP__ = VRP */
00177 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00178 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00179 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00180
00181 #elif defined(__open_xl__) && defined(__clang__)
00182 # define COMPILER_ID "IBMClang"
00183 # define COMPILER_VERSION_MAJOR DEC(__open_xl_version__)
00184 # define COMPILER_VERSION_MINOR DEC(__open_xl_release__)
00185 # define COMPILER_VERSION_PATCH DEC(__open_xl_modification__)
00186 # define COMPILER_VERSION_TWEAK DEC(__open_xl_ptf_fix_level__)
00187
00188
00189 #elif defined(__ibmxl__) && defined(__clang__)
00190 # define COMPILER_ID "XLClang"
00191 # define COMPILER_VERSION_MAJOR DEC(__ibmxl_version__)
00192 # define COMPILER_VERSION_MINOR DEC(__ibmxl_release__)
00193 # define COMPILER_VERSION_PATCH DEC(__ibmxl_modification__)
00194 # define COMPILER_VERSION_TWEAK DEC(__ibmxl_ptf_fix_level__)
00195
00196
00197 #elif defined(__IBMCPP__) && !defined(__COMPILER_VER__) && __IBMCPP__ >= 800
00198 # define COMPILER_ID "XL"
00199   /* __IBMCPP__ = VRP */
00200 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00201 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00202 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00203
00204 #elif defined(__IBMCPP__) && !defined(__COMPILER_VER__) && __IBMCPP__ < 800
00205 # define COMPILER_ID "VisualAge"
00206   /* __IBMCPP__ = VRP */
00207 # define COMPILER_VERSION_MAJOR DEC(__IBMCPP__/100)
00208 # define COMPILER_VERSION_MINOR DEC(__IBMCPP__/10 % 10)
00209 # define COMPILER_VERSION_PATCH DEC(__IBMCPP__    % 10)
00210
00211 #elif defined(__NVCOMPILER)
00212 # define COMPILER_ID "NVHPC"
00213 # define COMPILER_VERSION_MAJOR DEC(__NVCOMPILER_MAJOR__)
00214 # define COMPILER_VERSION_MINOR DEC(__NVCOMPILER_MINOR__)
```

```
00215 # if defined(__NVCOMPILER_PATCHLEVEL__)
00216 #  define COMPILER_VERSION_PATCH DEC(__NVCOMPILER_PATCHLEVEL__)
00217 # endif
00218
00219 #elif defined(__PGI)
00220 # define COMPILER_ID "PGI"
00221 # define COMPILER_VERSION_MAJOR DEC(__PGIC__)
00222 # define COMPILER_VERSION_MINOR DEC(__PGIC_MINOR__)
00223 # if defined(__PGIC_PATCHLEVEL__)
00224 #  define COMPILER_VERSION_PATCH DEC(__PGIC_PATCHLEVEL__)
00225 # endif
00226
00227 #elif defined(_CRAYC)
00228 # define COMPILER_ID "Cray"
00229 # define COMPILER_VERSION_MAJOR DEC(_RELEASE_MAJOR)
00230 # define COMPILER_VERSION_MINOR DEC(_RELEASE_MINOR)
00231
00232 #elif defined(__TI_COMPILER_VERSION__)
00233 # define COMPILER_ID "TI"
00234   /* __TI_COMPILER_VERSION__ = VVVRRRPPP */
00235 # define COMPILER_VERSION_MAJOR DEC(__TI_COMPILER_VERSION__/1000000)
00236 # define COMPILER_VERSION_MINOR DEC(__TI_COMPILER_VERSION__/1000   % 1000)
00237 # define COMPILER_VERSION_PATCH DEC(__TI_COMPILER_VERSION__        % 1000)
00238
00239 #elif defined(__CLANG_FUJITSU)
00240 # define COMPILER_ID "FujitsuClang"
00241 # define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00242 # define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00243 # define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00244 # define COMPILER_VERSION_INTERNAL_STR __clang_version__
00245
00246
00247 #elif defined(__FUJITSU)
00248 # define COMPILER_ID "Fujitsu"
00249 # if defined(__FCC_version__)
00250 #   define COMPILER_VERSION __FCC_version__
00251 # elif defined(__FCC_major__)
00252 #   define COMPILER_VERSION_MAJOR DEC(__FCC_major__)
00253 #   define COMPILER_VERSION_MINOR DEC(__FCC_minor__)
00254 #   define COMPILER_VERSION_PATCH DEC(__FCC_patchlevel__)
00255 # endif
00256 # if defined(__fcc_version)
00257 #   define COMPILER_VERSION_INTERNAL DEC(__fcc_version)
00258 # elif defined(__FCC_VERSION)
00259 #   define COMPILER_VERSION_INTERNAL DEC(__FCC_VERSION)
00260 # endif
00261
00262
00263 #elif defined(__ghs__)
00264 # define COMPILER_ID "GHS"
00265 /* __GHS_VERSION_NUMBER = VVVVRP */
00266 # ifdef __GHS_VERSION_NUMBER
00267 # define COMPILER_VERSION_MAJOR DEC(__GHS_VERSION_NUMBER / 100)
00268 # define COMPILER_VERSION_MINOR DEC(__GHS_VERSION_NUMBER / 10 % 10)
00269 # define COMPILER_VERSION_PATCH DEC(__GHS_VERSION_NUMBER      % 10)
00270 # endif
00271
00272 #elif defined(__TASKING__)
00273 # define COMPILER_ID "Tasking"
00274   # define COMPILER_VERSION_MAJOR DEC(__VERSION__/1000)
00275   # define COMPILER_VERSION_MINOR DEC(__VERSION__ % 100)
00276 # define COMPILER_VERSION_INTERNAL DEC(__VERSION__)
00277
00278 #elif defined(__SCO_VERSION__)
00279 # define COMPILER_ID "SCO"
00280
00281 #elif defined(__ARMCC_VERSION) && !defined(__clang__)
00282 # define COMPILER_ID "ARMCC"
00283 #if __ARMCC_VERSION >= 1000000
00284   /* __ARMCC_VERSION = VRRPPPP */
00285   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/1000000)
00286   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 100)
00287   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION     % 10000)
00288 #else
00289   /* __ARMCC_VERSION = VRPPPP */
00290   # define COMPILER_VERSION_MAJOR DEC(__ARMCC_VERSION/100000)
00291   # define COMPILER_VERSION_MINOR DEC(__ARMCC_VERSION/10000 % 10)
00292   # define COMPILER_VERSION_PATCH DEC(__ARMCC_VERSION    % 10000)
00293 #endif
00294
00295
00296 #elif defined(__clang__) && defined(__apple_build_version__)
00297 # define COMPILER_ID "AppleClang"
00298 # if defined(_MSC_VER)
00299 #  define SIMULATE_ID "MSVC"
00300 # endif
00301 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
```

```
00302 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00303 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00304 # if defined(_MSC_VER)
00305    /* _MSC_VER = VVRR */
00306 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00307 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00308 # endif
00309 # define COMPILER_VERSION_TWEAK DEC(__apple_build_version__)
00310
00311 #elif defined(__clang__) && defined(__ARMCOMPILER_VERSION)
00312 # define COMPILER_ID "ARMClang"
00313   # define COMPILER_VERSION_MAJOR DEC(__ARMCOMPILER_VERSION/1000000)
00314   # define COMPILER_VERSION_MINOR DEC(__ARMCOMPILER_VERSION/10000 % 100)
00315   # define COMPILER_VERSION_PATCH DEC(__ARMCOMPILER_VERSION     % 10000)
00316 # define COMPILER_VERSION_INTERNAL DEC(__ARMCOMPILER_VERSION)
00317
00318 #elif defined(__clang__)
00319 # define COMPILER_ID "Clang"
00320 # if defined(_MSC_VER)
00321 #  define SIMULATE_ID "MSVC"
00322 # endif
00323 # define COMPILER_VERSION_MAJOR DEC(__clang_major__)
00324 # define COMPILER_VERSION_MINOR DEC(__clang_minor__)
00325 # define COMPILER_VERSION_PATCH DEC(__clang_patchlevel__)
00326 # if defined(_MSC_VER)
00327    /* _MSC_VER = VVRR */
00328 #  define SIMULATE_VERSION_MAJOR DEC(_MSC_VER / 100)
00329 #  define SIMULATE_VERSION_MINOR DEC(_MSC_VER % 100)
00330 # endif
00331
00332 #elif defined(__LCC__) && (defined(__GNUC__) || defined(__GNUG__) || defined(__MCST__))
00333 # define COMPILER_ID "LCC"
00334 # define COMPILER_VERSION_MAJOR DEC(1)
00335 # if defined(__LCC__)
00336 #  define COMPILER_VERSION_MINOR DEC(__LCC__- 100)
00337 # endif
00338 # if defined(__LCC_MINOR__)
00339 #  define COMPILER_VERSION_PATCH DEC(__LCC_MINOR__)
00340 # endif
00341 # if defined(__GNUC__) && defined(__GNUC_MINOR__)
00342 #  define SIMULATE_ID "GNU"
00343 #  define SIMULATE_VERSION_MAJOR DEC(__GNUC__)
00344 #  define SIMULATE_VERSION_MINOR DEC(__GNUC_MINOR__)
00345 #  if defined(__GNUC_PATCHLEVEL__)
00346 #   define SIMULATE_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00347 #  endif
00348 # endif
00349
00350 #elif defined(__GNUC__) || defined(__GNUG__)
00351 # define COMPILER_ID "GNU"
00352 # if defined(__GNUC__)
00353 #  define COMPILER_VERSION_MAJOR DEC(__GNUC__)
00354 # else
00355 #  define COMPILER_VERSION_MAJOR DEC(__GNUG__)
00356 # endif
00357 # if defined(__GNUC_MINOR__)
00358 #  define COMPILER_VERSION_MINOR DEC(__GNUC_MINOR__)
00359 # endif
00360 # if defined(__GNUC_PATCHLEVEL__)
00361 #  define COMPILER_VERSION_PATCH DEC(__GNUC_PATCHLEVEL__)
00362 # endif
00363
00364 #elif defined(_MSC_VER)
00365 # define COMPILER_ID "MSVC"
00366   /* _MSC_VER = VVRR */
00367 # define COMPILER_VERSION_MAJOR DEC(_MSC_VER / 100)
00368 # define COMPILER_VERSION_MINOR DEC(_MSC_VER % 100)
00369 # if defined(_MSC_FULL_VER)
00370 #  if _MSC_VER >= 1400
00371     /* _MSC_FULL_VER = VVRRPPPPP */
00372 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 100000)
00373 #  else
00374     /* _MSC_FULL_VER = VVRRPPPP */
00375 #   define COMPILER_VERSION_PATCH DEC(_MSC_FULL_VER % 10000)
00376 #  endif
00377 # endif
00378 # if defined(_MSC_BUILD)
00379 #  define COMPILER_VERSION_TWEAK DEC(_MSC_BUILD)
00380 # endif
00381
00382 #elif defined(_ADI_COMPILER)
00383 # define COMPILER_ID "ADSP"
00384 #if defined(__VERSIONNUM__)
00385   /* __VERSIONNUM__ = 0xVVRRPPTT */
00386 #  define COMPILER_VERSION_MAJOR DEC(__VERSIONNUM__ » 24 & 0xFF)
00387 #  define COMPILER_VERSION_MINOR DEC(__VERSIONNUM__ » 16 & 0xFF)
00388 #  define COMPILER_VERSION_PATCH DEC(__VERSIONNUM__ » 8 & 0xFF)
```

```
00389 #  define COMPILER_VERSION_TWEAK DEC(__VERSIONNUM__ & 0xFF)
00390 #endif
00391
00392 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00393 # define COMPILER_ID "IAR"
00394 # if defined(__VER__) && defined(__ICCARM__)
00395 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 1000000)
00396 #  define COMPILER_VERSION_MINOR DEC(((__VER__) / 1000) % 1000)
00397 #  define COMPILER_VERSION_PATCH DEC((__VER__) % 1000)
00398 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00399 # elif defined(__VER__) && (defined(__ICCAVR__) || defined(__ICCRX__) || defined(__ICCRH850__) ||
      defined(__ICCRL78__) || defined(__ICC430__) || defined(__ICCRISCV__) || defined(__ICCV850__) ||
      defined(__ICC8051__) || defined(__ICCSTM8__))
00400 #  define COMPILER_VERSION_MAJOR DEC((__VER__) / 100)
00401 #  define COMPILER_VERSION_MINOR DEC((__VER__) - (((__VER__) / 100)*100))
00402 #  define COMPILER_VERSION_PATCH DEC(__SUBVERSION__)
00403 #  define COMPILER_VERSION_INTERNAL DEC(__IAR_SYSTEMS_ICC__)
00404 # endif
00405
00406
00407 /* These compilers are either not known or too old to define an
00408   identification macro.  Try to identify the platform and guess that
00409   it is the native compiler.  */
00410 #elif defined(__hpux) || defined(__hpua)
00411 # define COMPILER_ID "HP"
00412
00413 #else /* unknown compiler */
00414 # define COMPILER_ID ""
00415 #endif
00416
00417 /* Construct the string literal in pieces to prevent the source from
00418    getting matched.  Store it in a pointer rather than an array
00419    because some compilers will just produce instructions to fill the
00420    array rather than assigning a pointer to a static array.  */
00421 char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]";
00422 #ifdef SIMULATE_ID
00423 char const* info_simulate = "INFO" ":" "simulate[" SIMULATE_ID "]";
00424 #endif
00425
00426 #ifdef __QNXNTO__
00427 char const* qnxnto = "INFO" ":" "qnxnto[]";
00428 #endif
00429
00430 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00431 char const *info_cray = "INFO" ":" "compiler_wrapper[CrayPrgEnv]";
00432 #endif
00433
00434 #define STRINGIFY_HELPER(X) #X
00435 #define STRINGIFY(X) STRINGIFY_HELPER(X)
00436
00437 /* Identify known platforms by name.  */
00438 #if defined(__linux) || defined(__linux__) || defined(linux)
00439 # define PLATFORM_ID "Linux"
00440
00441 #elif defined(__MSYS__)
00442 # define PLATFORM_ID "MSYS"
00443
00444 #elif defined(__CYGWIN__)
00445 # define PLATFORM_ID "Cygwin"
00446
00447 #elif defined(__MINGW32__)
00448 # define PLATFORM_ID "MinGW"
00449
00450 #elif defined(__APPLE__)
00451 # define PLATFORM_ID "Darwin"
00452
00453 #elif defined(_WIN32) || defined(__WIN32__) || defined(WIN32)
00454 # define PLATFORM_ID "Windows"
00455
00456 #elif defined(__FreeBSD__) || defined(__FreeBSD)
00457 # define PLATFORM_ID "FreeBSD"
00458
00459 #elif defined(__NetBSD__) || defined(__NetBSD)
00460 # define PLATFORM_ID "NetBSD"
00461
00462 #elif defined(__OpenBSD__) || defined(__OPENBSD)
00463 # define PLATFORM_ID "OpenBSD"
00464
00465 #elif defined(__sun) || defined(sun)
00466 # define PLATFORM_ID "SunOS"
00467
00468 #elif defined(_AIX) || defined(__AIX) || defined(__AIX__) || defined(__aix) || defined(__aix__)
00469 # define PLATFORM_ID "AIX"
00470
00471 #elif defined(__hpux) || defined(__hpux__)
00472 # define PLATFORM_ID "HP-UX"
00473
```

```
00474 #elif defined(__HAIKU__)
00475 # define PLATFORM_ID "Haiku"
00476
00477 #elif defined(__BeOS) || defined(__BEOS__) || defined(_BEOS)
00478 # define PLATFORM_ID "BeOS"
00479
00480 #elif defined(__QNX__) || defined(__QNXNTO__)
00481 # define PLATFORM_ID "QNX"
00482
00483 #elif defined(__tru64) || defined(_tru64) || defined(__TRU64__)
00484 # define PLATFORM_ID "Tru64"
00485
00486 #elif defined(__riscos) || defined(__riscos__)
00487 # define PLATFORM_ID "RISCos"
00488
00489 #elif defined(__sinix) || defined(__sinix__) || defined(__SINIX__)
00490 # define PLATFORM_ID "SINIX"
00491
00492 #elif defined(__UNIX_SV__)
00493 # define PLATFORM_ID "UNIX_SV"
00494
00495 #elif defined(__bsdos__)
00496 # define PLATFORM_ID "BSDOS"
00497
00498 #elif defined(_MPRAS) || defined(MPRAS)
00499 # define PLATFORM_ID "MP-RAS"
00500
00501 #elif defined(__osf) || defined(__osf__)
00502 # define PLATFORM_ID "OSF1"
00503
00504 #elif defined(_SCO_SV) || defined(SCO_SV) || defined(sco_sv)
00505 # define PLATFORM_ID "SCO_SV"
00506
00507 #elif defined(__ultrix) || defined(__ultrix__) || defined(_ULTRIX)
00508 # define PLATFORM_ID "ULTRIX"
00509
00510 #elif defined(__XENIX__) || defined(_XENIX) || defined(XENIX)
00511 # define PLATFORM_ID "Xenix"
00512
00513 #elif defined(__WATCOMC__)
00514 # if defined(__LINUX__)
00515 #  define PLATFORM_ID "Linux"
00516
00517 # elif defined(__DOS__)
00518 #  define PLATFORM_ID "DOS"
00519
00520 # elif defined(__OS2__)
00521 #  define PLATFORM_ID "OS2"
00522
00523 # elif defined(__WINDOWS__)
00524 #  define PLATFORM_ID "Windows3x"
00525
00526 # elif defined(__VXWORKS__)
00527 #  define PLATFORM_ID "VxWorks"
00528
00529 # else /* unknown platform */
00530 #  define PLATFORM_ID
00531 # endif
00532
00533 #elif defined(__INTEGRITY)
00534 # if defined(INT_178B)
00535 #  define PLATFORM_ID "Integrity178"
00536
00537 # else /* regular Integrity */
00538 #  define PLATFORM_ID "Integrity"
00539 # endif
00540
00541 # elif defined(_ADI_COMPILER)
00542 #  define PLATFORM_ID "ADSP"
00543
00544 #else /* unknown platform */
00545 # define PLATFORM_ID
00546
00547 #endif
00548
00549 /* For windows compilers MSVC and Intel we can determine
00550    the architecture of the compiler being used.  This is because
00551    the compilers do not have flags that can change the architecture,
00552    but rather depend on which compiler is being used
00553 */
00554 #if defined(_WIN32) && defined(_MSC_VER)
00555 # if defined(_M_IA64)
00556 #  define ARCHITECTURE_ID "IA64"
00557
00558 # elif defined(_M_ARM64EC)
00559 #  define ARCHITECTURE_ID "ARM64EC"
00560
```

```
00561 # elif defined(_M_X64) || defined(_M_AMD64)
00562 #  define ARCHITECTURE_ID "x64"
00563
00564 # elif defined(_M_IX86)
00565 #  define ARCHITECTURE_ID "X86"
00566
00567 # elif defined(_M_ARM64)
00568 #  define ARCHITECTURE_ID "ARM64"
00569
00570 # elif defined(_M_ARM)
00571 #  if _M_ARM == 4
00572 #   define ARCHITECTURE_ID "ARMV4I"
00573 #  elif _M_ARM == 5
00574 #   define ARCHITECTURE_ID "ARMV5I"
00575 #  else
00576 #   define ARCHITECTURE_ID "ARMV" STRINGIFY(_M_ARM)
00577 #  endif
00578
00579 # elif defined(_M_MIPS)
00580 #  define ARCHITECTURE_ID "MIPS"
00581
00582 # elif defined(_M_SH)
00583 #  define ARCHITECTURE_ID "SHx"
00584
00585 # else /* unknown architecture */
00586 #  define ARCHITECTURE_ID ""
00587 # endif
00588
00589 #elif defined(__WATCOMC__)
00590 # if defined(_M_I86)
00591 #  define ARCHITECTURE_ID "I86"
00592
00593 # elif defined(_M_IX86)
00594 #  define ARCHITECTURE_ID "X86"
00595
00596 # else /* unknown architecture */
00597 #  define ARCHITECTURE_ID ""
00598 # endif
00599
00600 #elif defined(__IAR_SYSTEMS_ICC__) || defined(__IAR_SYSTEMS_ICC)
00601 # if defined(__ICCARM__)
00602 #  define ARCHITECTURE_ID "ARM"
00603
00604 # elif defined(__ICCRX__)
00605 #  define ARCHITECTURE_ID "RX"
00606
00607 # elif defined(__ICCRH850__)
00608 #  define ARCHITECTURE_ID "RH850"
00609
00610 # elif defined(__ICCRL78__)
00611 #  define ARCHITECTURE_ID "RL78"
00612
00613 # elif defined(__ICCRISCV__)
00614 #  define ARCHITECTURE_ID "RISCV"
00615
00616 # elif defined(__ICCAVR__)
00617 #  define ARCHITECTURE_ID "AVR"
00618
00619 # elif defined(__ICC430__)
00620 #  define ARCHITECTURE_ID "MSP430"
00621
00622 # elif defined(__ICCV850__)
00623 #  define ARCHITECTURE_ID "V850"
00624
00625 # elif defined(__ICC8051__)
00626 #  define ARCHITECTURE_ID "8051"
00627
00628 # elif defined(__ICCSTM8__)
00629 #  define ARCHITECTURE_ID "STM8"
00630
00631 # else /* unknown architecture */
00632 #  define ARCHITECTURE_ID ""
00633 # endif
00634
00635 #elif defined(__ghs__)
00636 # if defined(__PPC64__)
00637 #  define ARCHITECTURE_ID "PPC64"
00638
00639 # elif defined(__ppc__)
00640 #  define ARCHITECTURE_ID "PPC"
00641
00642 # elif defined(__ARM__)
00643 #  define ARCHITECTURE_ID "ARM"
00644
00645 # elif defined(__x86_64__)
00646 #  define ARCHITECTURE_ID "x64"
00647
```

```
00648 # elif defined(__i386__)
00649 #  define ARCHITECTURE_ID "X86"
00650
00651 # else /* unknown architecture */
00652 #  define ARCHITECTURE_ID ""
00653 # endif
00654
00655 #elif defined(__TI_COMPILER_VERSION__)
00656 # if defined(__TI_ARM__)
00657 #  define ARCHITECTURE_ID "ARM"
00658
00659 # elif defined(__MSP430__)
00660 #  define ARCHITECTURE_ID "MSP430"
00661
00662 # elif defined(__TMS320C28XX__)
00663 #  define ARCHITECTURE_ID "TMS320C28x"
00664
00665 # elif defined(__TMS320C6X__) || defined(_TMS320C6X)
00666 #  define ARCHITECTURE_ID "TMS320C6x"
00667
00668 # else /* unknown architecture */
00669 #  define ARCHITECTURE_ID ""
00670 # endif
00671
00672 # elif defined(__ADSPSHARC__)
00673 #  define ARCHITECTURE_ID "SHARC"
00674
00675 # elif defined(__ADSPBLACKFIN__)
00676 #  define ARCHITECTURE_ID "Blackfin"
00677
00678 #elif defined(__TASKING__)
00679
00680 # if defined(__CTC__) || defined(__CPTC__)
00681 #  define ARCHITECTURE_ID "TriCore"
00682
00683 # elif defined(__CMCS__)
00684 #  define ARCHITECTURE_ID "MCS"
00685
00686 # elif defined(__CARM__)
00687 #  define ARCHITECTURE_ID "ARM"
00688
00689 # elif defined(__CARC__)
00690 #  define ARCHITECTURE_ID "ARC"
00691
00692 # elif defined(__C51__)
00693 #  define ARCHITECTURE_ID "8051"
00694
00695 # elif defined(__CPCP__)
00696 #  define ARCHITECTURE_ID "PCP"
00697
00698 # else
00699 #  define ARCHITECTURE_ID ""
00700 # endif
00701
00702 #else
00703 #  define ARCHITECTURE_ID
00704 #endif
00705
00706 /* Convert integer to decimal digit literals.  */
00707 #define DEC(n)                   \
00708   ('0' + (((n) / 10000000)%10)), \
00709   ('0' + (((n) / 1000000)%10)),  \
00710   ('0' + (((n) / 100000)%10)),   \
00711   ('0' + (((n) / 10000)%10)),    \
00712   ('0' + (((n) / 1000)%10)),     \
00713   ('0' + (((n) / 100)%10)),      \
00714   ('0' + (((n) / 10)%10)),       \
00715   ('0' +  ((n) % 10))
00716
00717 /* Convert integer to hex digit literals.  */
00718 #define HEX(n)             \
00719   ('0' + ((n)>>28 & 0xF)), \
00720   ('0' + ((n)>>24 & 0xF)), \
00721   ('0' + ((n)>>20 & 0xF)), \
00722   ('0' + ((n)>>16 & 0xF)), \
00723   ('0' + ((n)>>12 & 0xF)), \
00724   ('0' + ((n)>>8  & 0xF)), \
00725   ('0' + ((n)>>4  & 0xF)), \
00726   ('0' + ((n)     & 0xF))
00727
00728 /* Construct a string literal encoding the version number. */
00729 #ifdef COMPILER_VERSION
00730 char const* info_version = "INFO" ":" "compiler_version[" COMPILER_VERSION "]";
00731
00732 /* Construct a string literal encoding the version number components. */
00733 #elif defined(COMPILER_VERSION_MAJOR)
00734 char const info_version[] = {
```

```
00735   'I', 'N', 'F', 'O', ':',
00736   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','[',
00737   COMPILER_VERSION_MAJOR,
00738 # ifdef COMPILER_VERSION_MINOR
00739   '.', COMPILER_VERSION_MINOR,
00740 #  ifdef COMPILER_VERSION_PATCH
00741    '.', COMPILER_VERSION_PATCH,
00742 #   ifdef COMPILER_VERSION_TWEAK
00743     '.', COMPILER_VERSION_TWEAK,
00744 #   endif
00745 #  endif
00746 # endif
00747   ']','\0'};
00748 #endif
00749
00750 /* Construct a string literal encoding the internal version number. */
00751 #ifdef COMPILER_VERSION_INTERNAL
00752 char const info_version_internal[] = {
00753   'I', 'N', 'F', 'O', ':',
00754   'c','o','m','p','i','l','e','r','_','v','e','r','s','i','o','n','_',
00755   'i','n','t','e','r','n','a','l','[',
00756   COMPILER_VERSION_INTERNAL,']','\0'};
00757 #elif defined(COMPILER_VERSION_INTERNAL_STR)
00758 char const* info_version_internal = "INFO" ":" "compiler_version_internal["
      COMPILER_VERSION_INTERNAL_STR "]";
00759 #endif
00760
00761 /* Construct a string literal encoding the version number components. */
00762 #ifdef SIMULATE_VERSION_MAJOR
00763 char const info_simulate_version[] = {
00764   'I', 'N', 'F', 'O', ':',
00765   's','i','m','u','l','a','t','e','_','v','e','r','s','i','o','n','[',
00766   SIMULATE_VERSION_MAJOR,
00767 # ifdef SIMULATE_VERSION_MINOR
00768   '.', SIMULATE_VERSION_MINOR,
00769 #  ifdef SIMULATE_VERSION_PATCH
00770    '.', SIMULATE_VERSION_PATCH,
00771 #   ifdef SIMULATE_VERSION_TWEAK
00772     '.', SIMULATE_VERSION_TWEAK,
00773 #   endif
00774 #  endif
00775 # endif
00776   ']','\0'};
00777 #endif
00778
00779 /* Construct the string literal in pieces to prevent the source from
00780    getting matched.  Store it in a pointer rather than an array
00781    because some compilers will just produce instructions to fill the
00782    array rather than assigning a pointer to a static array.  */
00783 char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]";
00784 char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]";
00785
00786
00787
00788 #if defined(__INTEL_COMPILER) && defined(_MSVC_LANG) && _MSVC_LANG < 201403L
00789 #  if defined(__INTEL_CXX11_MODE__)
00790 #    if defined(__cpp_aggregate_nsdmi)
00791 #      define CXX_STD 201402L
00792 #    else
00793 #      define CXX_STD 201103L
00794 #    endif
00795 #  else
00796 #    define CXX_STD 199711L
00797 #  endif
00798 #elif defined(_MSC_VER) && defined(_MSVC_LANG)
00799 #  define CXX_STD _MSVC_LANG
00800 #else
00801 #  define CXX_STD __cplusplus
00802 #endif
00803
00804 const char* info_language_standard_default = "INFO" ":" "standard_default["
00805 #if CXX_STD > 202002L
00806   "23"
00807 #elif CXX_STD > 201703L
00808   "20"
00809 #elif CXX_STD >= 201703L
00810   "17"
00811 #elif CXX_STD >= 201402L
00812   "14"
00813 #elif CXX_STD >= 201103L
00814   "11"
00815 #else
00816   "98"
00817 #endif
00818 "]";
00819
00820 const char* info_language_extensions_default = "INFO" ":" "extensions_default["
```

```
00821 #if (defined(__clang__) || defined(__GNUC__) || defined(__xlC__) ||           \
00822      defined(__TI_COMPILER_VERSION__)) &&                                     \
00823   !defined(__STRICT_ANSI__)
00824   "ON"
00825 #else
00826   "OFF"
00827 #endif
00828 "]";
00829
00830 /*--------------------------------------------------------------------------*/
00831
00832 int main(int argc, char* argv[])
00833 {
00834   int require = 0;
00835   require += info_compiler[argc];
00836   require += info_platform[argc];
00837   require += info_arch[argc];
00838 #ifdef COMPILER_VERSION_MAJOR
00839   require += info_version[argc];
00840 #endif
00841 #ifdef COMPILER_VERSION_INTERNAL
00842   require += info_version_internal[argc];
00843 #endif
00844 #ifdef SIMULATE_ID
00845   require += info_simulate[argc];
00846 #endif
00847 #ifdef SIMULATE_VERSION_MAJOR
00848   require += info_simulate_version[argc];
00849 #endif
00850 #if defined(__CRAYXT_COMPUTE_LINUX_TARGET)
00851   require += info_cray[argc];
00852 #endif
00853   require += info_language_standard_default[argc];
00854   require += info_language_extensions_default[argc];
00855   (void)argv;
00856   return require;
00857 }
```

6.9 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/BuildTeloc.cpp.o.d File Reference

6.10 BuildTeloc.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/BuildTeloc.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/BuildTeloc.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
```

```
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
```

```
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00154  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00155  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00156  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00157  /usr/include/c++/12/bits/fstream.tcc \
00158  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.11 build/default/CMakeFiles/ServiceTool.dir/BuildTeloc.cpp.o.d File Reference

6.12 BuildTeloc.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/BuildTeloc.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/BuildTeloc.cpp \
00003  /usr/include/stdc-predef.h \
00004  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00005  /usr/include/c++/12/iostream \
00006  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00007  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00008  /usr/include/features.h /usr/include/features-time64.h \
00009  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00010  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00011  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00012  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00013  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00014  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00015  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00016  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00017  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00018  /usr/include/c++/12/bits/stringfwd.h \
00019  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00020  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00021  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00022  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00023  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00024  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00025  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00026  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00032  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
```

```
00033  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00034  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00035  /usr/include/c++/12/bits/exception_ptr.h \
00036  /usr/include/c++/12/bits/exception_defines.h \
00037  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00038  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00039  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00040  /usr/include/c++/12/type_traits \
00041  /usr/include/c++/12/bits/nested_exception.h \
00042  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00043  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00044  /usr/include/x86_64-linux-gnu/bits/types.h \
00045  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00046  /usr/include/x86_64-linux-gnu/bits/time64.h \
00047  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00048  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00049  /usr/include/c++/12/bits/localefwd.h \
00050  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00051  /usr/include/c++/12/clocale /usr/include/locale.h \
00052  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00053  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00054  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00055  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00056  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00057  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00058  /usr/include/pthread.h /usr/include/sched.h \
00059  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00061  /usr/include/x86_64-linux-gnu/bits/sched.h \
00062  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00063  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00064  /usr/include/x86_64-linux-gnu/bits/time.h \
00065  /usr/include/x86_64-linux-gnu/bits/timex.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00070  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00071  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00073  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00074  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00075  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00076  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00077  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00078  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00079  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00080  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00081  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00082  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00083  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00084  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00085  /usr/include/c++/12/bits/allocator.h \
00086  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00087  /usr/include/c++/12/bits/new_allocator.h \
00088  /usr/include/c++/12/bits/functexcept.h \
00089  /usr/include/c++/12/bits/cpp_type_traits.h \
00090  /usr/include/c++/12/bits/ostream_insert.h \
00091  /usr/include/c++/12/bits/cxxabi_forced.h \
00092  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00093  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00094  /usr/include/c++/12/bits/concept_check.h \
00095  /usr/include/c++/12/debug/assertions.h \
00096  /usr/include/c++/12/bits/stl_iterator.h \
00097  /usr/include/c++/12/ext/type_traits.h \
00098  /usr/include/c++/12/bits/ptr_traits.h \
00099  /usr/include/c++/12/bits/stl_function.h \
00100  /usr/include/c++/12/backward/binders.h \
00101  /usr/include/c++/12/ext/numeric_traits.h \
00102  /usr/include/c++/12/bits/stl_algobase.h \
00103  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00104  /usr/include/c++/12/debug/debug.h \
00105  /usr/include/c++/12/bits/predefined_ops.h \
00106  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00107  /usr/include/c++/12/bits/range_access.h \
00108  /usr/include/c++/12/initializer_list \
00109  /usr/include/c++/12/bits/basic_string.h \
00110  /usr/include/c++/12/ext/alloc_traits.h \
00111  /usr/include/c++/12/bits/alloc_traits.h \
00112  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00113  /usr/include/c++/12/bits/functional_hash.h \
00114  /usr/include/c++/12/bits/string_view.tcc \
00115  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00116  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00117  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00118  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00119  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
```

```
00120  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00121  /usr/include/x86_64-linux-gnu/sys/select.h \
00122  /usr/include/x86_64-linux-gnu/bits/select.h \
00123  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00124  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00125  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00126  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00129  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00130  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00131  /usr/include/c++/12/cerrno /usr/include/errno.h \
00132  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00133  /usr/include/x86_64-linux-gnu/asm/errno.h \
00134  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00135  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00136  /usr/include/c++/12/bits/charconv.h \
00137  /usr/include/c++/12/bits/basic_string.tcc \
00138  /usr/include/c++/12/bits/locale_classes.tcc \
00139  /usr/include/c++/12/system_error \
00140  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00141  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00142  /usr/include/c++/12/bits/streambuf.tcc \
00143  /usr/include/c++/12/bits/basic_ios.h \
00144  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00145  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00147  /usr/include/c++/12/bits/streambuf_iterator.h \
00148  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00149  /usr/include/c++/12/bits/locale_facets.tcc \
00150  /usr/include/c++/12/bits/basic_ios.tcc \
00151  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00152  /usr/include/c++/12/bits/istream.tcc \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00155  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00156  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00157  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00158  /usr/include/c++/12/bits/fstream.tcc \
00159  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.13 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Compare.cpp.o.d File Reference

6.14 Compare.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Compare.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Compare.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
```

```
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
```

```
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Util.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Compare.h \
00155  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00156  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00157  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00158  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00159  /usr/include/c++/12/bits/fstream.tcc \
00160  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.15 build/default/CMakeFiles/ServiceTool.dir/Compare.cpp.o.d File Reference

6.16 Compare.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Compare.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Compare.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
```

```
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
```

```
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Util.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Compare.h \
00155  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00156  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00157  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00158  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00159  /usr/include/c++/12/bits/fstream.tcc \
00160  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.17 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Configuration.cpp.o.d File Reference

6.18 Configuration.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Configuration.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Configuration.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
```

```
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
```

```
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/c++/12/fstream \
00151  /usr/include/c++/12/bits/codecvt.h \
00152  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00153  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00154  /usr/include/c++/12/bits/fstream.tcc /usr/include/c++/12/vector \
00155  /usr/include/c++/12/bits/stl_uninitialized.h \
00156  /usr/include/c++/12/bits/stl_vector.h \
00157  /usr/include/c++/12/bits/stl_bvector.h \
00158  /usr/include/c++/12/bits/vector.tcc /usr/include/c++/12/sstream \
00159  /usr/include/c++/12/bits/sstream.tcc \
00160  /home/kali/libxl-4.2.0/include_cpp/libxl.h \
00161  /home/kali/libxl-4.2.0/include_cpp/IBookT.h \
00162  /home/kali/libxl-4.2.0/include_cpp/setup.h \
00163  /home/kali/libxl-4.2.0/include_cpp/enum.h \
00164  /home/kali/libxl-4.2.0/include_cpp/ISheetT.h \
00165  /home/kali/libxl-4.2.0/include_cpp/IFormatT.h \
00166  /home/kali/libxl-4.2.0/include_cpp/IFontT.h \
00167  /home/kali/libxl-4.2.0/include_cpp/IAutoFilterT.h \
00168  /home/kali/libxl-4.2.0/include_cpp/IFilterColumnT.h \
00169  /home/kali/libxl-4.2.0/include_cpp/IRichStringT.h \
00170  /home/kali/libxl-4.2.0/include_cpp/IFormControlT.h \
00171  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormatT.h \
00172  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormattingT.h \
00173  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00174  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00175  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00176  /home/kali/eclipse-workspace/ServiceTool/include/Configuration_impl.h \
00177  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00178  /home/kali/eclipse-workspace/ServiceTool/include/Compare.h \
00179  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00180  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.19 build/default/CMakeFiles/ServiceTool.dir/Configuration.cpp.o.d File Reference

6.20 Configuration.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Configuration.cpp.o: \
```

```
00002  /home/kali/eclipse-workspace/ServiceTool/Configuration.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
```

```
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/c++/12/fstream \
00151  /usr/include/c++/12/bits/codecvt.h \
00152  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00153  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00154  /usr/include/c++/12/bits/fstream.tcc \
00155  /home/kali/libxl-4.2.0/include_cpp/libxl.h \
00156  /home/kali/libxl-4.2.0/include_cpp/IBookT.h \
00157  /home/kali/libxl-4.2.0/include_cpp/setup.h \
00158  /home/kali/libxl-4.2.0/include_cpp/enum.h \
00159  /home/kali/libxl-4.2.0/include_cpp/ISheetT.h \
00160  /home/kali/libxl-4.2.0/include_cpp/IFormatT.h \
00161  /home/kali/libxl-4.2.0/include_cpp/IFontT.h \
00162  /home/kali/libxl-4.2.0/include_cpp/IAutoFilterT.h \
00163  /home/kali/libxl-4.2.0/include_cpp/IFilterColumnT.h \
00164  /home/kali/libxl-4.2.0/include_cpp/IRichStringT.h \
00165  /home/kali/libxl-4.2.0/include_cpp/IFormControlT.h \
00166  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormatT.h \
00167  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormattingT.h \
00168  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00169  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00170  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00171  /home/kali/eclipse-workspace/ServiceTool/include/Configuration_impl.h \
00172  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00173  /home/kali/eclipse-workspace/ServiceTool/include/Compare.h \
00174  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00175  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.21 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Configuration_impl.cpp.o.d File Reference

6.22 Configuration_impl.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Configuration_impl.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Configuration_impl.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
```

```
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/string.h \
00151  /usr/include/strings.h /usr/include/c++/12/vector \
00152  /usr/include/c++/12/bits/stl_uninitialized.h \
00153  /usr/include/c++/12/bits/stl_vector.h \
00154  /usr/include/c++/12/bits/stl_bvector.h \
00155  /usr/include/c++/12/bits/vector.tcc /usr/include/c++/12/sstream \
00156  /usr/include/c++/12/bits/sstream.tcc \
00157  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00158  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00159  /home/kali/libxl-4.2.0/include_cpp/libxl.h \
00160  /home/kali/libxl-4.2.0/include_cpp/IBookT.h \
00161  /home/kali/libxl-4.2.0/include_cpp/setup.h \
00162  /home/kali/libxl-4.2.0/include_cpp/enum.h \
```

```
00163  /home/kali/libxl-4.2.0/include_cpp/ISheetT.h \
00164  /home/kali/libxl-4.2.0/include_cpp/IFormatT.h \
00165  /home/kali/libxl-4.2.0/include_cpp/IFontT.h \
00166  /home/kali/libxl-4.2.0/include_cpp/IAutoFilterT.h \
00167  /home/kali/libxl-4.2.0/include_cpp/IFilterColumnT.h \
00168  /home/kali/libxl-4.2.0/include_cpp/IRichStringT.h \
00169  /home/kali/libxl-4.2.0/include_cpp/IFormControlT.h \
00170  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormatT.h \
00171  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormattingT.h \
00172  /home/kali/eclipse-workspace/ServiceTool/include/Configuration_impl.h \
00173  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00174  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00175  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00176  /usr/include/c++/12/bits/fstream.tcc \
00177  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00178  /home/kali/eclipse-workspace/ServiceTool/include/Util.h \
00179  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00180  /home/kali/eclipse-workspace/ServiceTool/include/Util.h \
00181  /home/kali/eclipse-workspace/ServiceTool/include/Compare.h \
00182  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00183  /home/kali/eclipse-workspace/ServiceTool/include/Teloc3000_Impl.h \
00184  /home/kali/eclipse-workspace/ServiceTool/include/Teloc4000_Impl.h
```

6.23 build/default/CMakeFiles/ServiceTool.dir/Configuration_impl.cpp.o.d File Reference

6.24 Configuration_impl.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Configuration_impl.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Configuration_impl.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
```

```
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
```

```
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/string.h \
00151  /usr/include/strings.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/libxl-4.2.0/include_cpp/libxl.h \
00155  /home/kali/libxl-4.2.0/include_cpp/IBookT.h \
00156  /home/kali/libxl-4.2.0/include_cpp/setup.h \
00157  /home/kali/libxl-4.2.0/include_cpp/enum.h \
00158  /home/kali/libxl-4.2.0/include_cpp/ISheetT.h \
00159  /home/kali/libxl-4.2.0/include_cpp/IFormatT.h \
00160  /home/kali/libxl-4.2.0/include_cpp/IFontT.h \
00161  /home/kali/libxl-4.2.0/include_cpp/IAutoFilterT.h \
00162  /home/kali/libxl-4.2.0/include_cpp/IFilterColumnT.h \
00163  /home/kali/libxl-4.2.0/include_cpp/IRichStringT.h \
00164  /home/kali/libxl-4.2.0/include_cpp/IFormControlT.h \
00165  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormatT.h \
00166  /home/kali/libxl-4.2.0/include_cpp/IConditionalFormattingT.h \
00167  /home/kali/eclipse-workspace/ServiceTool/include/Configuration_impl.h \
00168  /usr/include/c++/12/fstream /usr/include/c++/12/bits/codecvt.h \
00169  /usr/include/x86_64-linux-gnu/c++/12/bits/basic_file.h \
00170  /usr/include/x86_64-linux-gnu/c++/12/bits/c++io.h \
00171  /usr/include/c++/12/bits/fstream.tcc \
00172  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00173  /home/kali/eclipse-workspace/ServiceTool/include/Util.h \
00174  /home/kali/eclipse-workspace/ServiceTool/include/BuildTeloc.h \
00175  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.25 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Debug.cpp.o.d File Reference

6.26 Debug.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Debug.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Debug.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
```

```
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
```

```
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.27 build/default/CMakeFiles/ServiceTool.dir/Debug.cpp.o.d File Reference

6.28 Debug.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Debug.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Debug.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
```

```
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
```

```
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.29 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ ServiceTool.cpp.o.d File Reference

6.30 ServiceTool.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/ServiceTool.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/ServiceTool.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
```

```
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
```

```
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/build/cmake.debug.linux.x86_64/config.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.31 build/default/CMakeFiles/ServiceTool.dir/ServiceTool.cpp.o.d File Reference

6.32 ServiceTool.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/ServiceTool.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/ServiceTool.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
```

```
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
```

```
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/build/default/config.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Configuration.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.33 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Teloc3000_Impl.cpp.o.d File Reference

6.34 Teloc3000_Impl.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Teloc3000_Impl.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Teloc3000_Impl.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
```

```
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
```

```
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Teloc3000_Impl.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.35 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Teloc4000_Impl.cpp.o.d File Reference

6.36 Teloc4000_Impl.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Teloc4000_Impl.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Teloc4000_Impl.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
```

```
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
```

```
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc \
00151  /home/kali/eclipse-workspace/ServiceTool/include/Teloc4000_Impl.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Types.h
```

6.37 build/cmake.debug.linux.x86_64/CMakeFiles/ServiceTool.dir/↩ Util.cpp.o.d File Reference

6.38 Util.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Util.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Util.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
```

```
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
```

```
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/string.h \
00151  /usr/include/strings.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.39 build/default/CMakeFiles/ServiceTool.dir/Util.cpp.o.d File Reference

6.40 Util.cpp.o.d

Go to the documentation of this file.

```
00001 CMakeFiles/ServiceTool.dir/Util.cpp.o: \
00002  /home/kali/eclipse-workspace/ServiceTool/Util.cpp \
00003  /usr/include/stdc-predef.h /usr/include/c++/12/iostream \
00004  /usr/include/x86_64-linux-gnu/c++/12/bits/c++config.h \
00005  /usr/include/x86_64-linux-gnu/c++/12/bits/os_defines.h \
00006  /usr/include/features.h /usr/include/features-time64.h \
00007  /usr/include/x86_64-linux-gnu/bits/wordsize.h \
00008  /usr/include/x86_64-linux-gnu/bits/timesize.h \
00009  /usr/include/x86_64-linux-gnu/sys/cdefs.h \
00010  /usr/include/x86_64-linux-gnu/bits/long-double.h \
00011  /usr/include/x86_64-linux-gnu/gnu/stubs.h \
00012  /usr/include/x86_64-linux-gnu/gnu/stubs-64.h \
00013  /usr/include/x86_64-linux-gnu/c++/12/bits/cpu_defines.h \
00014  /usr/include/c++/12/pstl/pstl_config.h /usr/include/c++/12/ostream \
00015  /usr/include/c++/12/ios /usr/include/c++/12/iosfwd \
00016  /usr/include/c++/12/bits/stringfwd.h \
00017  /usr/include/c++/12/bits/memoryfwd.h /usr/include/c++/12/bits/postypes.h \
00018  /usr/include/c++/12/cwchar /usr/include/wchar.h \
00019  /usr/include/x86_64-linux-gnu/bits/libc-header-start.h \
00020  /usr/include/x86_64-linux-gnu/bits/floatn.h \
00021  /usr/include/x86_64-linux-gnu/bits/floatn-common.h \
00022  /usr/lib/gcc/x86_64-linux-gnu/12/include/stddef.h \
00023  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdarg.h \
00024  /usr/include/x86_64-linux-gnu/bits/wchar.h \
00025  /usr/include/x86_64-linux-gnu/bits/types/wint_t.h \
00026  /usr/include/x86_64-linux-gnu/bits/types/mbstate_t.h \
00027  /usr/include/x86_64-linux-gnu/bits/types/__mbstate_t.h \
00028  /usr/include/x86_64-linux-gnu/bits/types/__FILE.h \
00029  /usr/include/x86_64-linux-gnu/bits/types/FILE.h \
00030  /usr/include/x86_64-linux-gnu/bits/types/locale_t.h \
00031  /usr/include/x86_64-linux-gnu/bits/types/__locale_t.h \
00032  /usr/include/c++/12/exception /usr/include/c++/12/bits/exception.h \
00033  /usr/include/c++/12/bits/exception_ptr.h \
00034  /usr/include/c++/12/bits/exception_defines.h \
00035  /usr/include/c++/12/bits/cxxabi_init_exception.h \
00036  /usr/include/c++/12/typeinfo /usr/include/c++/12/bits/hash_bytes.h \
00037  /usr/include/c++/12/new /usr/include/c++/12/bits/move.h \
00038  /usr/include/c++/12/type_traits \
00039  /usr/include/c++/12/bits/nested_exception.h \
00040  /usr/include/c++/12/bits/char_traits.h /usr/include/c++/12/cstdint \
00041  /usr/lib/gcc/x86_64-linux-gnu/12/include/stdint.h /usr/include/stdint.h \
00042  /usr/include/x86_64-linux-gnu/bits/types.h \
00043  /usr/include/x86_64-linux-gnu/bits/typesizes.h \
00044  /usr/include/x86_64-linux-gnu/bits/time64.h \
00045  /usr/include/x86_64-linux-gnu/bits/stdint-intn.h \
00046  /usr/include/x86_64-linux-gnu/bits/stdint-uintn.h \
00047  /usr/include/c++/12/bits/localefwd.h \
00048  /usr/include/x86_64-linux-gnu/c++/12/bits/c++locale.h \
00049  /usr/include/c++/12/clocale /usr/include/locale.h \
00050  /usr/include/x86_64-linux-gnu/bits/locale.h /usr/include/c++/12/cctype \
00051  /usr/include/ctype.h /usr/include/x86_64-linux-gnu/bits/endian.h \
00052  /usr/include/x86_64-linux-gnu/bits/endianness.h \
00053  /usr/include/c++/12/bits/ios_base.h /usr/include/c++/12/ext/atomicity.h \
00054  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr.h \
00055  /usr/include/x86_64-linux-gnu/c++/12/bits/gthr-default.h \
00056  /usr/include/pthread.h /usr/include/sched.h \
00057  /usr/include/x86_64-linux-gnu/bits/types/time_t.h \
00058  /usr/include/x86_64-linux-gnu/bits/types/struct_timespec.h \
00059  /usr/include/x86_64-linux-gnu/bits/sched.h \
00060  /usr/include/x86_64-linux-gnu/bits/types/struct_sched_param.h \
00061  /usr/include/x86_64-linux-gnu/bits/cpu-set.h /usr/include/time.h \
```

```
00062  /usr/include/x86_64-linux-gnu/bits/time.h \
00063  /usr/include/x86_64-linux-gnu/bits/timex.h \
00064  /usr/include/x86_64-linux-gnu/bits/types/struct_timeval.h \
00065  /usr/include/x86_64-linux-gnu/bits/types/clock_t.h \
00066  /usr/include/x86_64-linux-gnu/bits/types/struct_tm.h \
00067  /usr/include/x86_64-linux-gnu/bits/types/clockid_t.h \
00068  /usr/include/x86_64-linux-gnu/bits/types/timer_t.h \
00069  /usr/include/x86_64-linux-gnu/bits/types/struct_itimerspec.h \
00070  /usr/include/x86_64-linux-gnu/bits/pthreadtypes.h \
00071  /usr/include/x86_64-linux-gnu/bits/thread-shared-types.h \
00072  /usr/include/x86_64-linux-gnu/bits/pthreadtypes-arch.h \
00073  /usr/include/x86_64-linux-gnu/bits/atomic_wide_counter.h \
00074  /usr/include/x86_64-linux-gnu/bits/struct_mutex.h \
00075  /usr/include/x86_64-linux-gnu/bits/struct_rwlock.h \
00076  /usr/include/x86_64-linux-gnu/bits/setjmp.h \
00077  /usr/include/x86_64-linux-gnu/bits/types/__sigset_t.h \
00078  /usr/include/x86_64-linux-gnu/bits/types/struct___jmp_buf_tag.h \
00079  /usr/include/x86_64-linux-gnu/bits/pthread_stack_min-dynamic.h \
00080  /usr/include/x86_64-linux-gnu/c++/12/bits/atomic_word.h \
00081  /usr/include/x86_64-linux-gnu/sys/single_threaded.h \
00082  /usr/include/c++/12/bits/locale_classes.h /usr/include/c++/12/string \
00083  /usr/include/c++/12/bits/allocator.h \
00084  /usr/include/x86_64-linux-gnu/c++/12/bits/c++allocator.h \
00085  /usr/include/c++/12/bits/new_allocator.h \
00086  /usr/include/c++/12/bits/functexcept.h \
00087  /usr/include/c++/12/bits/cpp_type_traits.h \
00088  /usr/include/c++/12/bits/ostream_insert.h \
00089  /usr/include/c++/12/bits/cxxabi_forced.h \
00090  /usr/include/c++/12/bits/stl_iterator_base_types.h \
00091  /usr/include/c++/12/bits/stl_iterator_base_funcs.h \
00092  /usr/include/c++/12/bits/concept_check.h \
00093  /usr/include/c++/12/debug/assertions.h \
00094  /usr/include/c++/12/bits/stl_iterator.h \
00095  /usr/include/c++/12/ext/type_traits.h \
00096  /usr/include/c++/12/bits/ptr_traits.h \
00097  /usr/include/c++/12/bits/stl_function.h \
00098  /usr/include/c++/12/backward/binders.h \
00099  /usr/include/c++/12/ext/numeric_traits.h \
00100  /usr/include/c++/12/bits/stl_algobase.h \
00101  /usr/include/c++/12/bits/stl_pair.h /usr/include/c++/12/bits/utility.h \
00102  /usr/include/c++/12/debug/debug.h \
00103  /usr/include/c++/12/bits/predefined_ops.h \
00104  /usr/include/c++/12/bits/refwrap.h /usr/include/c++/12/bits/invoke.h \
00105  /usr/include/c++/12/bits/range_access.h \
00106  /usr/include/c++/12/initializer_list \
00107  /usr/include/c++/12/bits/basic_string.h \
00108  /usr/include/c++/12/ext/alloc_traits.h \
00109  /usr/include/c++/12/bits/alloc_traits.h \
00110  /usr/include/c++/12/bits/stl_construct.h /usr/include/c++/12/string_view \
00111  /usr/include/c++/12/bits/functional_hash.h \
00112  /usr/include/c++/12/bits/string_view.tcc \
00113  /usr/include/c++/12/ext/string_conversions.h /usr/include/c++/12/cstdlib \
00114  /usr/include/stdlib.h /usr/include/x86_64-linux-gnu/bits/waitflags.h \
00115  /usr/include/x86_64-linux-gnu/bits/waitstatus.h \
00116  /usr/include/x86_64-linux-gnu/sys/types.h /usr/include/endian.h \
00117  /usr/include/x86_64-linux-gnu/bits/byteswap.h \
00118  /usr/include/x86_64-linux-gnu/bits/uintn-identity.h \
00119  /usr/include/x86_64-linux-gnu/sys/select.h \
00120  /usr/include/x86_64-linux-gnu/bits/select.h \
00121  /usr/include/x86_64-linux-gnu/bits/types/sigset_t.h \
00122  /usr/include/alloca.h /usr/include/x86_64-linux-gnu/bits/stdlib-float.h \
00123  /usr/include/c++/12/bits/std_abs.h /usr/include/c++/12/cstdio \
00124  /usr/include/stdio.h /usr/include/x86_64-linux-gnu/bits/types/__fpos_t.h \
00125  /usr/include/x86_64-linux-gnu/bits/types/__fpos64_t.h \
00126  /usr/include/x86_64-linux-gnu/bits/types/struct_FILE.h \
00127  /usr/include/x86_64-linux-gnu/bits/types/cookie_io_functions_t.h \
00128  /usr/include/x86_64-linux-gnu/bits/stdio_lim.h \
00129  /usr/include/c++/12/cerrno /usr/include/errno.h \
00130  /usr/include/x86_64-linux-gnu/bits/errno.h /usr/include/linux/errno.h \
00131  /usr/include/x86_64-linux-gnu/asm/errno.h \
00132  /usr/include/asm-generic/errno.h /usr/include/asm-generic/errno-base.h \
00133  /usr/include/x86_64-linux-gnu/bits/types/error_t.h \
00134  /usr/include/c++/12/bits/charconv.h \
00135  /usr/include/c++/12/bits/basic_string.tcc \
00136  /usr/include/c++/12/bits/locale_classes.tcc \
00137  /usr/include/c++/12/system_error \
00138  /usr/include/x86_64-linux-gnu/c++/12/bits/error_constants.h \
00139  /usr/include/c++/12/stdexcept /usr/include/c++/12/streambuf \
00140  /usr/include/c++/12/bits/streambuf.tcc \
00141  /usr/include/c++/12/bits/basic_ios.h \
00142  /usr/include/c++/12/bits/locale_facets.h /usr/include/c++/12/cwctype \
00143  /usr/include/wctype.h /usr/include/x86_64-linux-gnu/bits/wctype-wchar.h \
00144  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_base.h \
00145  /usr/include/c++/12/bits/streambuf_iterator.h \
00146  /usr/include/x86_64-linux-gnu/c++/12/bits/ctype_inline.h \
00147  /usr/include/c++/12/bits/locale_facets.tcc \
00148  /usr/include/c++/12/bits/basic_ios.tcc \
```

```
00149  /usr/include/c++/12/bits/ostream.tcc /usr/include/c++/12/istream \
00150  /usr/include/c++/12/bits/istream.tcc /usr/include/string.h \
00151  /usr/include/strings.h \
00152  /home/kali/eclipse-workspace/ServiceTool/include/Debug.h \
00153  /home/kali/eclipse-workspace/ServiceTool/include/Types.h \
00154  /home/kali/eclipse-workspace/ServiceTool/include/Util.h
```

6.41 build/cmake.debug.linux.x86_64/config.h File Reference

Macros

- #define ServiceTool_VERSION_MAJOR 0
- #define ServiceTool_VERSION_MINOR 2

6.41.1 Macro Definition Documentation

6.41.1.1 ServiceTool_VERSION_MAJOR

```
#define ServiceTool_VERSION_MAJOR 0
```

Definition at line 1 of file config.h.

6.41.1.2 ServiceTool_VERSION_MINOR

```
#define ServiceTool_VERSION_MINOR 2
```

Definition at line 2 of file config.h.

6.42 config.h

Go to the documentation of this file.

```
00001 #define ServiceTool_VERSION_MAJOR 0
00002 #define ServiceTool_VERSION_MINOR 2
```

6.43 build/default/config.h File Reference

Macros

- #define ServiceTool_VERSION_MAJOR 0
- #define ServiceTool_VERSION_MINOR 2

6.43.1 Macro Definition Documentation

6.43.1.1 ServiceTool_VERSION_MAJOR

```
#define ServiceTool_VERSION_MAJOR 0
```

Definition at line 1 of file config.h.

6.43.1.2 ServiceTool_VERSION_MINOR

```
#define ServiceTool_VERSION_MINOR 2
```

Definition at line 2 of file config.h.

6.44 config.h

Go to the documentation of this file.

```
00001 #define ServiceTool_VERSION_MAJOR 0
00002 #define ServiceTool_VERSION_MINOR 2
```

6.45 build/cmake.debug.linux.x86_64/detect_compiler_builtins.cpp File Reference

6.46 detect_compiler_builtins.cpp

Go to the documentation of this file.

6.47 build/default/detect_compiler_builtins.cpp File Reference

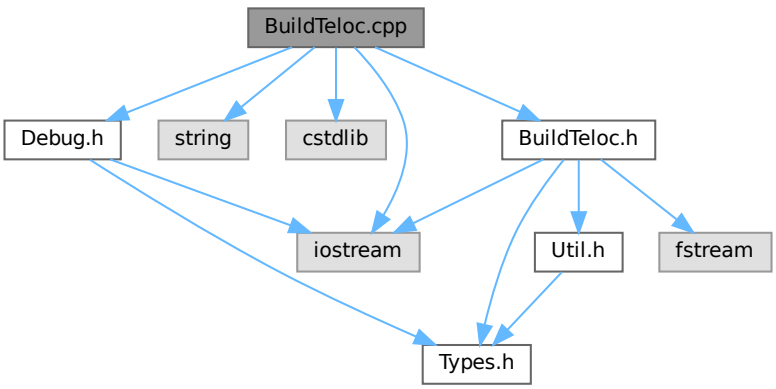
6.48 detect_compiler_builtins.cpp

Go to the documentation of this file.

6.49 BuildTeloc.cpp File Reference

```
#include <iostream>
#include <string>
#include <cstdlib>
#include "Debug.h"
#include "BuildTeloc.h"
```

Include dependency graph for BuildTeloc.cpp:



Macros

- #define MAX_BOARD_1500 17
- #define MAX_BOARD_2500 8

Functions

- type_::UINT64 lookuptableposition (const std::string name)

- void lookuptableTeloc1500 (const type_::CHAR *const s, type_::UINT64 index, std::ofstream &myfile)
  the function build the teloc 1500
- void lookuptableTeloc2500 (const type_::CHAR *const s, type_::UINT64 index, std::ofstream &myfile)
  the function build the teloc 2500
- type_::ebool lookuptablefamily (const std::string family, std::string &name)
  !!!

Variables

- ofstream myfile
- buildteloc_::t_buildtelocstruct buildtelocstruct [TELOC_BOARD][TELOC_BOARD]

6.49.1 Macro Definition Documentation

6.49.1.1 MAX_BOARD_1500

```
#define MAX_BOARD_1500 17
```

Definition at line 12 of file BuildTeloc.cpp.

6.49.1.2 MAX_BOARD_2500

```
#define MAX_BOARD_2500 8
```

Definition at line 13 of file BuildTeloc.cpp.

6.49.2 Function Documentation

6.49.2.1 lookuptablefamily()

```
type_::ebool lookuptablefamily (
            const std::string family,
            std::string & name )
```

!!!

the function check the family code for the boards

Parameters

family,value	to check in the database
name,the	board name linked to the family code

Returns

board name

Definition at line 142 of file BuildTeloc.cpp.

6.49.2.2 lookuptableposition()

```
type_::UINT64 lookuptableposition (
            const std::string name )
```

Definition at line 162 of file BuildTeloc.cpp.

6.49.2.3 lookuptableTeloc1500()

```
void lookuptableTeloc1500 (
            const type_::CHAR *const s,
            type_::UINT64 index,
            std::ofstream & myfile )
```

the function build the teloc 1500

Parameters

∗s,pointer	to be string read in xlsx file

Returns

Definition at line 66 of file BuildTeloc.cpp.

6.49.2.4 lookuptableTeloc2500()

```
void lookuptableTeloc2500 (
            const type_::CHAR *const s,
            type_::UINT64 index,
            std::ofstream & myfile )
```

the function build the teloc 2500

Parameters

∗s,pointer	to be string read in xlsx file

Returns

Definition at line 104 of file BuildTeloc.cpp.

6.49.3 Variable Documentation

6.49.3.1 buildtelocstruct

```
buildteloc_::t_buildtelocstruct buildtelocstruct[TELOC_BOARD][TELOC_BOARD]
```

Definition at line 19 of file BuildTeloc.cpp.

6.49.3.2 myfile

```
ofstream myfile
```

Definition at line 16 of file BuildTeloc.cpp.

6.50 BuildTeloc.cpp

Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include <string>
00008 #include <cstdlib>
00009 #include "Debug.h"
00010 #include"BuildTeloc.h"
00011
00012 #define MAX_BOARD_1500 17
00013 #define MAX_BOARD_2500 8
00014
00015
00016 ofstream myfile;
00017
00018
00019 buildteloc_::t_buildtelocstruct buildtelocstruct[TELOC_BOARD][TELOC_BOARD];
00020
00021 static buildteloc_::t_buildtelocstruct *getaccesstelocbuild(void)
00022 {
00023     static buildteloc_::t_buildtelocstruct buildtelocstruct;
00024     return(&buildtelocstruct);
00025 }
00026
00027 static buildteloc_::t_teloc_config *getteloccconfig(void)
00028 {
00029     static buildteloc_::t_teloc_config teloc_config;
00030     return(&teloc_config);
00031 }
00032
00033
00034 static type_::UINT64 lookuptable_board(std::string board_name)
00035 {
00036     type_::UINT64 lret = -1U;
00037     static const std::string table_board[] = {"POSU", "IOCO", "CORE", "REBO", "MVB",
      "CPM","DAIO","SRAM","SABOA","BAPLB","BAPLI","BUPLB","MAINC","PC104","FLASH"};
```

```
00038     for(type_::UINT64 index = 0;index < TELOC_BOARD; index++)
00039     {
00040         if(table_board[index] == board_name)
00041         {
00042             lret = index;
00043             break;
00044         }
00045     }
00046     return(lret);
00047 }
00048
00049 static void setboard(std::string board_name, buildteloc_::t_buildtelocstruct *ptr)
00050 {
00051     if(util_::CheckArg((buildteloc_::t_buildtelocstruct*) ptr) == type_::RESULT_OK)
00052     {
00053     //  std::cout<<FUNCTION_NAME<<std::endl;
00054         ptr->board_name = board_name;
00055         ptr->active = type_::TRUE;
00056         ptr->numberofboard++;
00057     }
00058 }
00059
00066 void lookuptableTeloc1500(const type_::CHAR * const s, type_::UINT64 index, std::ofstream &myfile)
00067 {
00068     // configure the database
00069     static std::string table[MAX_BOARD_1500][MAX_BOARD_1500] = DATABASE_BOARD_T1500;
00070     if(util_::CheckArg((type_::CHAR*) s) == type_::RESULT_OK)
00071     {
00072         std::cout<<FUNCTION_NAME<<std::endl;
00073         type_::UINT64 ii = 0;
00074         for(ii = 0; ii < MAX_BOARD_1500; ii++)
00075         {
00076             if(table[ii][0] == s)
00077             {
00078                 std::cout<<table[ii][0]<< " == ";
00079                 std::cout<<table[ii][1]<<std::endl;
00080                 myfile <<"the "<<table[ii][1]<<" "<<table[ii][0]<<" is present"<<endl;
00081                 setboard(
00082                 (table[ii][1]),
00083                 &buildtelocstruct[index][lookuptable_board(table[ii][1])]
00084                 );
00085                 DEBUG_ENABLE(debug_::enable)
00086                 {
00087                     std::cout<<"name == ";
00088
      std::cout<<buildtelocstruct[index][lookuptable_board(table[ii][1])].board_name<<std::endl;
00089                     std::cout<<"active == ";
00090
      std::cout<<buildtelocstruct[index][lookuptable_board(table[ii][1])].active<<std::endl;
00091                 }
00092                 break;
00093                 // set
00094             }//if
00095         }//for
00096     }
00097 }
00104 void lookuptableTeloc2500(const type_::CHAR * const s, type_::UINT64 index, std::ofstream &myfile)
00105 {
00106     static std::string table[MAX_BOARD_2500][MAX_BOARD_2500] = DATABASE_BOARD_T2500;
00107     if(util_::CheckArg((type_::CHAR*) s) == type_::RESULT_OK)
00108     {
00109 /*  std::string ss = s;
00110     std::cout<<ss<<endl;
00111     ss = ss.substr(0,10);
00112     std::cout<<ss<<endl;
00113     the engine could read the string without the version!!!!
00114 */
00115         //std::cout<<FUNCTION_NAME<<std::endl;
00116         for(int ii = 0; ii < MAX_BOARD_2500; ii++)
00117         {
00118             if(table[ii][0] == s)
00119             {
00120                 std::cout<<table[ii][0]<< " == ";
00121                 std::cout<<table[ii][1]<<std::endl;
00122                 myfile <<"the "<<table[ii][1]<<" "<<table[ii][0]<<" is present"<<endl;
00123                 setboard(
00124                 (table[ii][1]),
00125                 &buildtelocstruct[index][lookuptable_board(table[ii][1])]
00126                 );
00127                 DEBUG_ENABLE(debug_::enable)
00128                 {
00129                     std::cout<<"name == ";
00130
      std::cout<<buildtelocstruct[index][lookuptable_board(table[ii][1])].board_name<<std::endl;
00131                     std::cout<<"active == ";
00132
      std::cout<<buildtelocstruct[index][lookuptable_board(table[ii][1])].active<<std::endl;
```

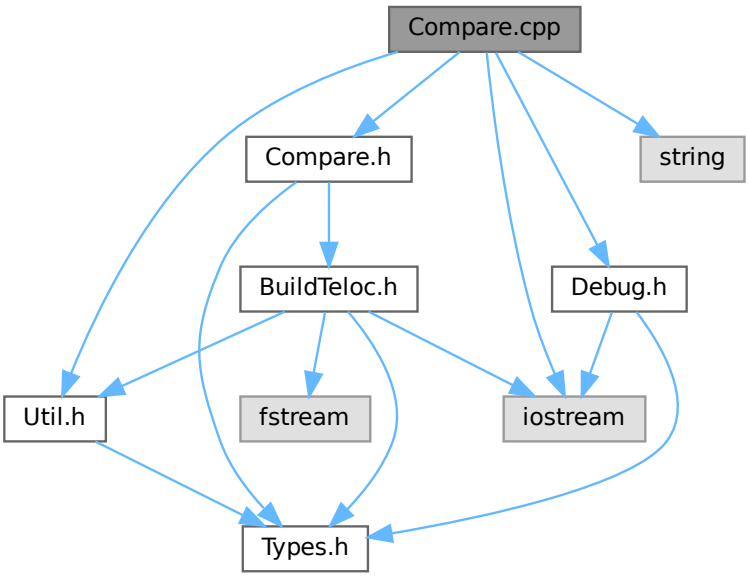
```
00133                 }
00134                 break;
00135                 // set
00136             }//if
00137         }//for
00138     }
00139 }
00141
00142 type_::ebool lookuptablefamily(const std::string family, std::string &name)
00143 {
00144     static std::string table[DATABASE_FAMILY_TX500_SIZE][DATABASE_FAMILY_TX500_SIZE] =
      DATABASE_FAMILY_TX500;
00145     type_::ebool lfind = type_::FALSE;
00146     for(int jj = 0; jj < DATABASE_FAMILY_TX500_SIZE; jj++)
00147     {
00148         //std::cout«"Table = "«table[jj][0]«std::endl;
00149         //std::cout«"family = "«family«std::endl;
00150         if(table[jj][0] == family)
00151         {
00152             lfind = type_::TRUE;
00153             name = table[jj][1];
00154             //std::cout<getline(osheet, line);<"name = "«name«std::endl;
00155             break;
00156         }//if
00157     }//for
00158     //std::cout«"lfind = "«lfind«std::endl;
00159     return(lfind);
00160 }
00161
00162 type_::UINT64 lookuptableposition(const std::string name)
00163 {
00164     static std::string table[POSITION_TO_WRITING_SIZE][POSITION_TO_WRITING_SIZE] =
      POSITION_TO_WRITING;
00165     type_::UINT64 lpos = 0x0U;
00166     for(type_::UINT64 jj = 0; jj < POSITION_TO_WRITING_SIZE; jj++)
00167     {
00168         if(table[jj][0] == name)
00169         {
00170             lpos = atoi(table[jj][1].c_str());
00171             break;
00172         }
00173     }
00174     return(lpos);
00175 }
00176
```

6.51 Compare.cpp File Reference

in this file are implmented the methods used to comapre the dirrent telocs read

```
#include <iostream>
#include <string>
#include "Debug.h"
#include "Util.h"
#include "Compare.h"
```

Include dependency graph for Compare.cpp:



Macros

- #define TABEL_SIZE 12

Functions

- type_::UINT64 plausibilitycheck_numberboard (buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[ ][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
- type_::UINT64 plausibilitycheck_boards (buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[ ][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
- type_::ebool compare_handle (type_::UINT64 main_config, type_::UINT64 code_config)

  the function manages the main compare

6.51.1 Detailed Description

in this file are implmented the methods used to comapre the dirrent telocs read

Author

Salvatore Muoio

Definition in file Compare.cpp.

6.51.2 Macro Definition Documentation

6.51.2.1 TABEL_SIZE

```
#define TABEL_SIZE 12
```

Definition at line 13 of file Compare.cpp.

6.51.3 Function Documentation

6.51.3.1 compare_handle()

```
type_::ebool compare_handle (
            type_::UINT64 main_config,
            type_::UINT64 code_config )
```

the function manages the main compare

Parameters

main_config,main	configuration
code_config	configuration code

Returns

value of not match

Definition at line 119 of file Compare.cpp.

6.51.3.2 plausibilitycheck_boards()

```
type_::UINT64 plausibilitycheck_boards (
            buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[][TELOC_BOARD],
            type_::UINT64 len,
            type_::UINT64 jj )
```

Definition at line 88 of file Compare.cpp.

6.51.3.3 plausibilitycheck_numberboard()

```
type_::UINT64 plausibilitycheck_numberboard (
            buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[][TELOC_BOARD],
            type_::UINT64 len,
            type_::UINT64 jj )
```

Definition at line 61 of file Compare.cpp.

6.52 Compare.cpp

Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include <string>
00008 #include "Debug.h"
00009 #include "Util.h"
00010 #include "Compare.h"
00011
00012
00013 #define TABEL_SIZE 12
00014
00015
00022 static type_::ebool isTelocUUC(type_::UINT64 index)
00023 {
00024     return((index == 0) ? type_::TRUE : type_::FALSE);
00025 }
00026
00034 static type_::ebool isendloop(type_::UINT64 index, type_::UINT64 len)
00035 {
00036     return((index >= len-1) ? type_::TRUE : type_::FALSE);
00037 }
00038
00039 static type_::UINT64  lookuptablevaluemacth(const std::string board)
00040 {
00041     type_::UINT64 lvalue = 0x0U;
00042     static std::string table[TABEL_SIZE][TABEL_SIZE] = TABLE_MATCH_VALUE;
00043     for(type_::UINT64 ii = 0; ii <TABEL_SIZE;  ii++)
00044     {
00045         if(table[ii][0] == board)
00046         {
00047             lvalue = stoi(table[ii][1]);
00048             break;
00049         }//if
00050     }//for
00051     return(lvalue);
00052 }
00053
00054 static type_::UINT64 match(std::string board)
00055 {
00056     type_::UINT64 lmatch = 0x0U;
00057     lmatch = lookuptablevaluemacth(board);
00058     return(lmatch);
00059 }
00060
00061 type_::UINT64 plausibilitycheck_numberboard(buildteloc_::t_buildtelocstruct
      ptrbuildtelocstruc[][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
00062 {
00063     type_::UINT64 lCnt = 0;
00064 //  if(util_::CheckArg((buildteloc_::t_buildtelocstruct*) ptrbuildtelocstruc) == type_::RESULT_OK)
00065     {
00066         std::cout«std::endl;
00067         for(type_::UINT64 ii = 0; ii< len; ii++){
00068
00069         //while(!ptrbuildtelocstruc){
00070             if(ptrbuildtelocstruc[jj][ii].active == type_::TRUE)
00071             {
00072                 if(debug_::enable)
00073                 {
00074                     std::cout«lCnt«" board active = "«ptrbuildtelocstruc[jj][ii].active«std::endl;
00075                     std::cout«lCnt«" board name = "«ptrbuildtelocstruc[jj][ii].board_name«std::endl;
00076                     std::cout«lCnt«" number of board =
      "«ptrbuildtelocstruc[jj][ii].numberofboard«std::endl;
00077                 }
00078                 lCnt = lCnt + ptrbuildtelocstruc[jj][ii].numberofboard;
00079             }//if
00080         }//for
00081         //}//while
00082         std::cout«"Teloc["«jj«"] under check:: number of boards = "«lCnt«std::endl;
00083     }//body
00084     return(lCnt);
00085 }
00086
00087
00088 type_::UINT64 plausibilitycheck_boards(buildteloc_::t_buildtelocstruct
      ptrbuildtelocstruc[][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
00089 {
00090     type_::ebool lMatch = type_::TRUE;
00091     type_::UINT64 valuematch = 0x00U;
00092     if(isTelocUUC(jj) == type_::FALSE)
00093     {
00094         std::cout«"plausibiity check boards"«std::endl;
00095         for(type_::UINT64 ii = 0; ii < len; ii++)
00096             for(type_::UINT64 tt = 0; tt < len; tt++)
```

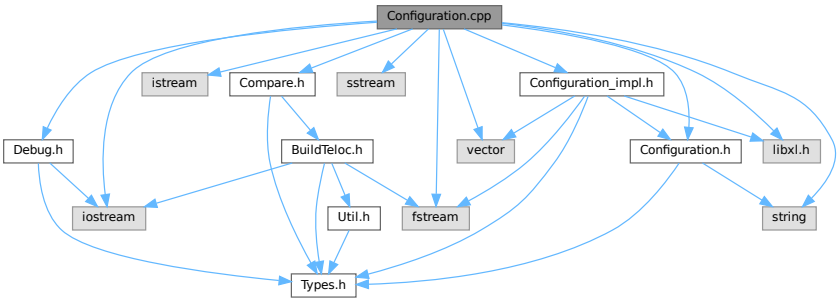
```
00097             {
00098                 if(ptrbuildtelocstruc[jj][ii].active == type_::TRUE)
00099                 {
00100                     if(ptrbuildtelocstruc[jj][ii].board_name == ptrbuildtelocstruc[0][tt].board_name)
00101                     {
00102                         std::cout« "Board match == "«ptrbuildtelocstruc[jj][ii].board_name«std::endl;
00103                         break;
00104                     }//if
00105                     else if(isendloop(tt, len) == type_::TRUE)
00106                     {
00107                         std::cout« "Board not matched ==
      "«ptrbuildtelocstruc[jj][ii].board_name«std::endl;
00108                         lMatch = type_::FALSE;
00109                         std::cout«"match =="«match(ptrbuildtelocstruc[jj][ii].board_name)«std::endl;
00110                         valuematch -= match(ptrbuildtelocstruc[jj][ii].board_name);
00111                         std::cout«"valuematch =="«valuematch«std::endl;
00112                     }//Elseif
00113                 }//if
00114             }//for
00115     }//if
00116     return(valuematch);
00117 }
00118
00119 type_::ebool compare_handle(type_::UINT64 main_config, type_::UINT64 code_config)
00120 {
00121     type_::ebool match = type_:: FALSE;
00122     const type_::UINT64 basic_config = 0x11E0;
00123     std::cout«"main_config =="«main_config«std::endl;
00124     std::cout«"code_config =="«code_config«std::endl;
00125     if(code_config == main_config)//both code are equal
00126         match = type_::TRUE;
00127     else
00128     {
00129         // calculate the xor or both and verify if it-s included in the basic_config
00130         type_::UINT64 lresult = main_config ^ code_config;
00131         lresult &= basic_config;
00132         if(lresult == basic_config)
00133             match = type_::TRUE;
00134     }
00135     std::cout«"'match ='"«match«std::endl;
00136     return(match);
00137 }
00138
00139
```

6.53 Configuration.cpp File Reference

in this file are implmented the methods used to work with Configuration file

```
#include <iostream>
#include <fstream>
#include <istream>
#include <vector>
#include <sstream>
#include <string>
#include "libxl.h"
#include "Debug.h"
#include "Configuration.h"
#include "Configuration_impl.h"
#include "Compare.h"
```

Include dependency graph for Configuration.cpp:



Functions

- ifstream fname ("Config.ini", ios_base::in)

6.53.1 Detailed Description

in this file are implmented the methods used to work with Configuration file

Author

Salvatore Muoio

Definition in file Configuration.cpp.

6.53.2 Function Documentation

6.53.2.1 fname()

```
ifstream fname (
             "Config.ini" ,
             ios_base::in  )
```

6.54 Configuration.cpp

Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include <fstream>
00008 #include <istream>
00009 #include <vector>
00010 #include <sstream>
00011 #include <string>
00012 #include "libxl.h"
00013 #include "Debug.h"
00014 #include "Configuration.h"
00015 #include "Configuration_impl.h"
00016 #include "Compare.h"
```

```
00017
00018 using namespace libxl;
00019 using namespace std;
00020 //using namespace debug_;
00021
00022
00023 ifstream fname("Config.ini", ios_base::in);
00024
00025
00026 config::config()
00027 {
00028     // constructor
00029     //std::cout<<FUNCTION_NAME<<std::endl;
00030     pimpl = new(configimpl);
00031 }
00032
00033
00034 config_::t_configstruct * config::getconfigstruct(void)
00035 {
00036     static config_::t_configstruct configstruct;
00037     return(&configstruct);
00038 }
00039
00040 config_::t_telocstrcut * config::gettelocstruct(void)
00041 {
00042     static config_::t_telocstrcut telocstruct;
00043     return(&telocstruct);
00044 }
00045
00046 config &config::getinstance()
00047 {
00048     // TODO: insert return statement here
00049     static config instance;
00050     std::cout<<"getinstance"<<std::endl;
00051     return instance;
00052 }
00053
00054
00055
00056 void config::readfileconfig(void)
00057 {
00058     //read configuration file
00059    //std::string line;
00060 /*    Book* book  = xlCreateXMLBook();
00061    Book* obook = xlCreateXMLBook();*/
00062    type_::UINT64 number_of_Teloc = 0x0U;
00063    std::string name_matrix;
00064    std::ofstream MatrixTeloc;//("Teloc_Matrix.csv");
00065    if(fname.is_open()){
00066    getline(fname, getconfigstruct()->line);
00067    std::cout<<"line :: "<<getconfigstruct()->line<<std::endl;
00068   gettelocstruct()->kindofTeloc = whoamI(getconfigstruct()->line);
00069   std::cout<<"kindofTeloc ==: "<<gettelocstruct()->kindofTeloc<<std::endl;
00070    pimpl->extract_filename(getconfigstruct()->line, getconfigstruct()->filename,
00071                            &getconfigstruct()->assemblycode[number_of_Teloc][0]);
00072    name_matrix = getconfigstruct()->line.substr(15, 4);
00073
00074    name_matrix = "Teloc_Matrix_"+name_matrix+".csv";
00075    std::cout<<"name_matrix = "<<name_matrix<<std::endl;
00076    ofstream MatrixTeloc(name_matrix);
00077    pimpl->create_template(MatrixTeloc, gettelocstruct()->kindofTeloc);
00078
00079     ifstream Data(getconfigstruct()->filename, ifstream::in);
00080   if(Data.is_open()){
00081       std::cout<<"Enter in the loop"<<std::endl;
00082       getline(Data, getconfigstruct()->title);
00083
00084       while(getline(Data, getconfigstruct()->line))
00085             //getline(Data, getconfigstruct()->line);
00086       {
00087                 //std::cout<<"line = "<<getconfigstruct()->line<<std::endl;
00088                 std::stringstream rowStream(getconfigstruct()->line);
00089                 std::string draft;
00090                 std::vector<std::string> cols;
00091                 while(getline(rowStream, draft, ';')){
00092             //getconfigstruct()->assemblyteloc[ii++] = draft;
00093                 cols.push_back(draft);
00094                 }
00095                 pimpl->create_output_file(cols, MatrixTeloc);
00096                 //getconfigstruct()->index_row++;
00097         }
00098   }
00099   }
00100   MatrixTeloc.close();
00101   std::cout<<"index_row ="<<getconfigstruct()->index_row<<std::endl;
00102   //open the file in reading mode
00103   fstream CompareTeloc(name_matrix, ios::in|ios::out);
```

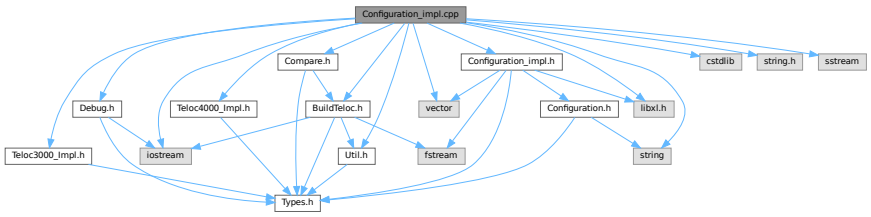
```
00104   if(CompareTeloc.is_open())
00105   pimpl->compare_create_configuration(CompareTeloc);
00106   else
00107       std::cout«"file not open"«std::endl;
00108
00109 }
00110
00111 static string findteloccode(std::string line)
00112 {
00113     static char key = '_';
00114     std::string teloccode;
00115     std::cout«FUNCTION_NAME«std::endl;
00116     size_t pos = line.find(key);
00117     teloccode = line.substr(pos+1, (4));
00118     //DEBUG_DISPLAY(debug_::enable, teloccode);
00119     std::cout«"teloc = "«teloccode«std::endl;
00120     return(teloccode);
00121 }
00122
00123 static string lookuptable(std::string teloccode)
00124 {
00125      std::string lret = "unknown";
00126      static std::string table[2][2] = {{"1500","T1500"},
00127                                        {"2500","T2500"}};
00128
00129     for(type_::UINT8 jj = 0; jj < 2; jj++)
00130     {
00131         if(table[jj][0] == teloccode)
00132         {
00133             lret = table[jj][1];
00134             break;
00135         }//if
00136     }//for
00137     return(lret);
00138 }
00139
00140 string config::whoamI(std::string line)
00141 {
00142     std::string teloc = findteloccode(line);
00143     std::cout«FUNCTION_NAME«std::endl;
00144     //DEBUG_DISPLAY(debug_::enable, teloc);
00145     return(lookuptable(teloc));
00146 }
```

6.55 Configuration_impl.cpp File Reference

in this file are implemented the methods used to work with Configuration file

```
#include <iostream>
#include <string>
#include <cstdlib>
#include <string.h>
#include <vector>
#include <sstream>
#include "Debug.h"
#include "libxl.h"
#include "Configuration_impl.h"
#include "Util.h"
#include "BuildTeloc.h"
#include "Compare.h"
#include "Teloc3000_Impl.h"
#include "Teloc4000_Impl.h"
```

Include dependency graph for Configuration_impl.cpp:



Functions

- configimpl_::t_configstructimpl * getaccescfgimpl (void)
- configimpl_::t_filestruct * getfilestruct (void)
- void closefile (void)

Variables

- void(* lookuptableTeloc [2])(const type_::CHAR *const s, type_::UINT64 index, std::ofstream &myfile) = {&lookuptableTeloc1500, &lookuptableTeloc2500}
- Sheet * sheet

6.55.1 Detailed Description

in this file are implemented the methods used to work with Configuration file

Author

Salvatore Muoio

Definition in file Configuration_impl.cpp.

6.55.2 Function Documentation

6.55.2.1 closefile()

```
void closefile (
            void  )
```

Definition at line 46 of file Configuration_impl.cpp.

6.55.2.2 getaccescfgimpl()

```
configimpl_::t_configstructimpl * getaccescfgimpl (
            void  )
```

Definition at line 34 of file Configuration_impl.cpp.

6.55.2.3 getfilestruct()

```
configimpl_::t_filestruct * getfilestruct (
            void  )
```

Definition at line 40 of file Configuration_impl.cpp.

6.55.3 Variable Documentation

6.55.3.1 lookuptableTeloc

```
void(* lookuptableTeloc[2])(const type_::CHAR *const s, type_::UINT64 index, std::ofstream
&myfile) (
            const type_::CHAR *const s,
            type_::UINT64 index,
            std::ofstream & myfile ) = {&lookuptableTeloc1500, &lookuptableTeloc2500}
```

Definition at line 30 of file Configuration_impl.cpp.

6.55.3.2 sheet

```
Sheet* sheet
```

Definition at line 32 of file Configuration_impl.cpp.

6.56 Configuration_impl.cpp

Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include <string>
00008 #include <cstdlib>
00009 #include <string.h>
00010 #include <vector>
00011 #include <sstream>
00012 #include "Debug.h"
00013 #include "libxl.h"
00014 #include "Configuration_impl.h"
00015 #include "Util.h"
00016 #include "BuildTeloc.h"
00017 #include "Compare.h"
00018 #include "Teloc3000_Impl.h"
00019 #include "Teloc4000_Impl.h"
00020
00021 using namespace libxl;
00022 //using namespace teloc3000impl_;
00023
00024 using namespace std;
00025 static ofstream myfile;
00026 static ofstream fCompare;
00027
00028 static void reset_filestruct(configimpl_::t_filestruct *ptr);
00029 static void write_file(ofstream &file, configimpl_::t_filestruct *ptr);
00030 void (*lookuptableTeloc[2])(const type_::CHAR * const s, type_::UINT64 index, std::ofstream &myfile) =
      {&lookuptableTeloc1500, &lookuptableTeloc2500};
00031
00032 Sheet *sheet;
00033
00034 configimpl_::t_configstructimpl *getaccescfgimpl(void)
00035 {
00036     static configimpl_::t_configstructimpl configstructimpl;
00037     return(&configstructimpl);
00038 }
```

```
00039
00040 configimpl_::t_filestruct *getfilestruct(void)
00041 {
00042     static configimpl_::t_filestruct filestructimpl;
00043     return(&filestructimpl);
00044 }
00045
00046 void closefile(void)
00047 {
00048     myfile.close();
00049 }
00050
00051 static type_::ebool checkmaincode(type_::UINT64 maincode)
00052 {
00053     return ((maincode != 0) ? (type_::TRUE) : (type_::FALSE));
00054 }
00055
00056
00057
00058 void config::configimpl::extract_filename(std::string line, type_::CHAR *filename, type_::CHAR*
    assemblycode)
00059 {
00060     //DEBUG_INFO(1);
00061     static char key = ':';
00062     static type_::UINT64 len = 13;
00063     std::cout<<FUNCTION_NAME<<std::endl;
00064     std::cout<<line.length()<<std::endl;
00065     size_t pos = line.find(key);
00066     //cout<<"pos:"<<pos<<endl;
00067     //DEBUG_DISPLAY(debug_::enable, pos);
00068     //debug_::dbg_display(pos);
00069     std::cout<<"line :: "<<line<<std::endl;
00070     memcpy(filename, &line.at(pos+1), line.length());
00071     //filename = name;
00072     //DEBUG_DISPLAY(debug_::enable, filename);
00073     memcpy(assemblycode, &line.at(pos+1), 13);
00074
00075 }
00076
00077 void config::configimpl::extract_column_compare(std::string line, type_::CHAR *col)
00078 {
00079     std::cout<<FUNCTION_NAME<<std::endl;
00080     if(col)
00081     {
00082         static char key = ':';
00083         size_t pos = line.find(key);
00084         DEBUG_DISPLAY(debug_::enable, pos);
00085         memcpy(col, &line.at(pos+1), line.length());
00086         //*col = line.at(pos+1);
00087         DEBUG_DISPLAY(debug_::enable, col);
00088     }
00089 }
00090
00091 type_::ebool config::configimpl::find_column(const type_::CHAR *title,  const type_::CHAR *col)
00092 {
00093     type_::ebool ret = type_::FALSE;
00094     std::cout<<FUNCTION_NAME<<std::endl;
00095     if ( strcmp( title, col ) == 0 )
00096     {
00097         ret = type_::TRUE;
00098         DEBUG_DISPLAY(debug_::enable, title);
00099         DEBUG_DISPLAY(debug_::enable, col);
00100     }
00101     return(ret);
00102 }
00103
00104 type_::ebool config::configimpl::parser_kenfile(const type_:: CHAR *col, std::string filename)
00105 {
00106     type_::ebool find = type_::FALSE;
00107 //  static type_::UINT64 column = 0x0U;
00108     std::cout<<FUNCTION_NAME<<std::endl;
00109     std::cout<<"parser_kenfile::filename == "<<filename<<std::endl;
00110     Book *book = xlCreateXMLBook();
00111     if(book->load(filename.c_str()))
00112     {
00113         std::cout<<"file xlsx"<<std::endl;
00114         //set the right column where are all teloc components
00115         // open sheet 0
00116         sheet = book->getSheet(0);
00117         for(type_::UINT64 jj = sheet->firstCol(); jj < sheet->lastCol(); ++jj)
00118         {
00119             CellType cellType = sheet->cellType(0, jj);
00120             const char* s = sheet->readStr(0, jj);
00121             std::cout << (s ? s : "null") << " [string]"<<std::endl;
00122             if(find_column(s, col) == type_::TRUE)
00123             {
00124                 //set colucreate_main_configmn!!!
```

```
00125                 getaccescfgimpl()->findcolumn = jj;
00126                 DEBUG_DISPLAY(debug_::enable, getaccescfgimpl()->findcolumn);
00127         //      column = jj;
00128             //  DEBUG_DISPLAY(debug_::enable, jj);
00129                 find = type_::TRUE;
00130                 break;
00131             }
00132         }//for
00133         //parser_kenfile
00134     }
00135     else
00136         std::cout«"not file xlsx"«std::endl;
00137     return(find);
00138 }
00139
00140 void config::configimpl::scroll_column(const std::string telocode)
00141 {
00143     static type_::UINT64 index = 0;
00144     std::cout«FUNCTION_NAME«std::endl;
00145     DEBUG_DISPLAY(debug_::enable, getaccescfgimpl()->findcolumn);
00146     std::cout«"telocode == "«telocode«std::endl;
00147     std::cout«"convert == "«util_::ConverTelocCode2Num(telocode)«std::endl;
00148     myfile.open ("Teloc.txt",ios::app);
00149     if(myfile.is_open())
00150     {
00151         myfile« "Teloc is "«telocode;
00152         myfile«"  "«sheet->name()«endl;
00153         for(type_::UINT64 row = sheet->firstRow()+1; row < sheet->lastRow(); row++)
00154         {
00155         //
00156             CellType cellType = sheet->cellType(row, getaccescfgimpl()->findcolumn);
00157             const char* s = sheet->readStr(row, getaccescfgimpl()->findcolumn);
00158             //std::cout « (s ? s : "null") « " [string]"«std::endl;
00159             // check the code in the string with data for the teloc
00160             lookuptableTeloc[util_::ConverTelocCode2Num(telocode)](s, index, myfile);
00161
00162         }//for
00163     }//if
00164     index++;
00165 }
00166
00167 type_::UINT64 config::configimpl::getsizeTeloc(void)
00168 {
00169     return(TELOC_BOARD);
00170 }
00171
00172 static type_::UINT64 check_kind_teloc(std::string teloc)
00173 {
00174     return(teloc == "T2500" ? 2 : 1);
00175 }
00176
00177 static void create_template_1500(ofstream &osheet)
00178 {
00179     const char* table[FAMILY_TELOC_1500_SIZE] = FAMILY_TELOC_1500;
00180     const std::string separator = ";";
00181     char *s;
00182     if(osheet.is_open())
00183     {
00184         osheet«"Asswmbly Code";
00185         osheet«separator;
00186         osheet«"Customer";
00187         osheet«separator;
00188         for(type_::UINT64 ii = 0; ii < FAMILY_TELOC_1500_SIZE; ii++)
00189         {
00190             //std::cout« "table["«ii«"] = "«table[ii]«std::endl;
00191             osheet«table[ii];
00192             osheet«separator;
00193         }//for
00194         osheet«endl;
00195     }//if
00196
00197 }
00198
00199 static void create_template_2500(ofstream &osheet)
00200 {
00201     const char* table[FAMILY_TELOC_2500_SIZE] = FAMILY_TELOC_2500;
00202     const std::string separator = ";";
00203     char *s;
00204     if(osheet.is_open())
00205     {
00206         osheet«"Asswmbly Code";
00207         osheet«separator;
00208         osheet«"Customer";
00209         osheet«separator;
00210         for(type_::UINT64 ii = 0; ii < FAMILY_TELOC_2500_SIZE; ii++)
00211         {
00212             //std::cout« "table["«ii«"] = "«table[ii]«std::endl;
```

```
00213             osheet<<table[ii];
00214             osheet<<separator;
00215         }//for
00216         osheet<<endl;
00217     }//if
00218
00219 }
00220
00221
00222
00223
00224 void config::configimpl::create_template(ofstream &osheet, std::string teloc)
00225 {
00226     //static type_::UINT64 size = 0x0U;
00227     switch(check_kind_teloc(teloc))
00228     {
00229     case 1:
00230         std::cout<<"teloc 1500"<<std::endl;
00231         create_template_1500(osheet);
00232         break;
00233     case 2:
00234         std::cout<<"teloc 2500"<<std::endl;
00235         create_template_2500(osheet);
00236         break;
00237     }
00238     // write the family
00239 }
00240
00241 void config::configimpl::create_teloc_assembly(const char *s, Sheet *osheet, type_::UINT64 row)
00242 {
00243     static char assembly_code[13] = {0};
00244     {
00245         osheet->writeStr(row+1, 0, s);
00246     }//if
00247
00248 }
00249
00250 void write_file(ofstream &file, configimpl_::t_filestruct *ptr)
00251 {
00252
00253     const std::string separator = ";";
00254     if(file.is_open())
00255     {
00256         file<<ptr->assembly_code;
00257         file<<separator;
00258         file<<ptr->customer;
00259         file<<separator;
00260         file<<ptr->posu;
00261         file<<separator;
00262         file<<ptr->core;
00263         file<<separator;
00264         file<<ptr->ioco;
00265         file<<separator;
00266         file<<ptr->daio;
00267         file<<separator;
00268         file<<ptr->rebo;
00269         file<<separator;
00270         file<<ptr->sabo;
00271         file<<separator;
00272         file<<ptr->mvb;
00273         file<<separator;
00274         file<<ptr->can;
00275         file<<separator;
00276         file<<ptr->gps;
00277         file<<separator;
00278         file<<ptr->cpm;
00279         file<<separator;
00280         file<<ptr->sram;
00281         file<<separator;
00282         file<<ptr->flash;
00283         file<<separator;
00284         file<<ptr->backplane;
00285         file<<separator;
00286         file<<ptr->datra;
00287         file<<separator;
00288         file<<endl;
00289     }
00290     reset_filestruct(ptr);
00291 }
00292
00293 void reset_filestruct(configimpl_::t_filestruct *ptr)
00294 {
00295     ptr->assembly_code = "";
00296     ptr->customer = "";
00297     ptr->posu = "";
00298     ptr->core = "";
00299     ptr->ioco = "";
```

```
00300     ptr->daio = "";
00301     ptr->rebo = "";
00302     ptr->sabo = "";
00303     ptr->mvb = "";
00304     ptr->can = "";
00305     ptr->gps = "";
00306     ptr->cpm = "";
00307     ptr->sram = "";
00308     ptr->flash = "";
00309     ptr->backplane = "";
00310     ptr->datra = "";
00311 }
00312
00313
00314 void config::configimpl::create_output_file(std::vector<std::string> col, ofstream &file)
00315 {
00316     //std::vector<std::string> row;
00317     const type_::UINT64 aasembly_size = 13;
00318     static std::string assembly = "";
00319     //static type_::ebool toogle = type_::FALSE;
00320     //column = col;
00321     //std::cout<<"function create_output_file "<<std::endl;
00322     std::cout<<col[0]<<std::endl;
00323     std::cout<<assembly<<std::endl;
00324     if((col[0] != "") && (col[0].size()) == aasembly_size)
00325     {
00326         if(col[0] != assembly)
00327         {
00328         //toogle = type_::TRUE;
00329             getconfigstruct()->index_row++;
00330             std::cout<<"assembly = "<<assembly<<std::endl;
00331             std::cout<<"col[0] = "<<col[0]<<std::endl;
00332             std::cout<<"col[4] = "<<col[4]<<std::endl;
00333             //column.push_back(col[0]);
00334             assembly = col[0];
00335             write_file(file, getfilestruct());
00336             getfilestruct()->assembly_code = assembly;
00337             //add customer in thes trcture
00338             getfilestruct()->customer = col[4];
00339         }
00340         std::cout<<"col[3] = "<<col[3]<<std::endl;
00341             extract_family(col[3], file, getfilestruct());
00342     }
00343
00344 }
00345
00346 void config::configimpl::extract_version(std::string code, std::string &variant)
00347 {
00348     variant = code.substr(10,3);
00349     std::cout<<"extract_version::variant = "<<variant<<std::endl;
00350
00351 }
00352
00353
00354
00355 void config::configimpl::extract_family(std::string code, ofstream &osheet,configimpl_::t_filestruct
      *ptr)
00356 {
00357     std::string family;
00358     std::string variant;
00359     std::string name;
00360     type_::UINT64 pos = 0x0U;
00361     family = code.substr(5,5);
00362     std::cout<<"family = "<<family<<std::endl;
00363     // verify
00364     //lookuptablefamily(family);
00365     if(lookuptablefamily(family, name) == type_::TRUE)
00366     {
00367         // extract the version
00368         extract_version(code, variant);
00369         pos = lookuptableposition(name);
00370         std::cout<<"pos = "<<pos<<std::endl;
00371         if(pos > 0)
00372         {
00373             //Check the right position
00374             std::string lcode = code.substr(7,7);
00375             std::cout<<"extract_family::variant = "<<variant<<std::endl;
00376             write_variant(pos, lcode, ptr);
00377             //osheet<<variant;
00378             //write variant in the file!!!
00379             // write customer
00380         }
00381     }
00382 }
00383
00384 void config::configimpl::write_variant( type_::UINT64 pos, std::string variant,
      configimpl_::t_filestruct *ptr)
```

```
00385 {
00386     const std::string common = ", ";
00387     std::string tmp = "";
00388     tmp = variant;
00389     switch(pos)
00390     {
00391         case 3:
00392             ptr->posu += tmp + common;
00393         break;
00394         case 4:
00395             ptr->core += tmp + common;
00396         break;
00397         case 5:
00398             ptr->ioco += tmp + common;
00399         break;
00400         case 6:
00401             ptr->daio += tmp + common;
00402         break;
00403         case 7:
00404             ptr->rebo += tmp + common;
00405         break;
00406         case 8:
00407             ptr->sabo += tmp + common;
00408         break;
00409         case 9:
00410             ptr->mvb += tmp + common;
00411         break;
00412         case 10:
00413             ptr->can += tmp + common;
00414         break;
00415         case 11:
00416             ptr->gps += tmp + common;
00417         break;
00418         case 12:
00419             ptr->cpm += tmp + common;
00420         break;
00421         case 13:
00422             ptr->sram += tmp + common;
00423         break;
00424         case 14:
00425             ptr->flash += tmp + common;
00426         break;
00427         case 15:
00428             ptr->backplane += tmp + common;
00429         break;
00430         case 16:
00431             ptr->datra += tmp + common;
00432         break;
00433         default:
00434             break;
00435     }
00436     tmp = "";
00437 }
00438
00439 static vector<string> create_row(std::string line)
00440 {
00441     vector<string> lrow;
00442     //type_::UINT64  ii = 0;
00443     std::string word;
00444     //std::cout«"create_row line = "«line«std::endl;
00445     //getline()
00446     lrow.clear();
00447     stringstream s(line);
00448     while(getline(s, word, ';'))
00449     {
00450         lrow.push_back(word);
00451         //std::cout«"word("«ii«") = "«word«std::endl;
00452         //ii++;
00453     }
00454     //std::cout«"!!!!! end create_row line !!!!!!"«std::endl;
00455     return(lrow);
00456 }
00457
00458 static type_::UINT64 create_code(std::vector<string> row)
00459 {
00460     type_::UINT64 lcode = 0x0U;
00461     //std::cout«"vec size = "«(int)row.size«std::endl;
00462     if(row.at(0) != "")
00463     {
00464         for(type_::UINT64 jj = 2; jj < FAMILY_TELOC_1500_SIZE; jj++)
00465         {
00466             //std::cout«"jj = "«jj«std::endl;
00467             //std::cout«"row = "«row.at(jj)«std::endl;
00468             if (row.at(jj) != "")
00469             {
00470                 lcode += 1«((jj)-2);
00471                 //std::cout«"lcode = "«lcode«std::endl;
```

```
00472             }
00473         }
00474     }
00475     //std::cout«"lcode ="«lcode«std::endl;
00476     //std::cout«"lcode = "«lcode«std::endl;
00477     return(lcode&MASK_CODE);
00478 }
00479
00480
00481 static type_::UINT64 create_main_config(std::string line)
00482 {
00483     //--->>
00484 //  type_::UINT64 main_code = 0x0U;
00485     //vector<string> row;
00486     type_::UINT64 codeconfig = 0x0U;
00487     //row = create_row(line);
00488     //std::cout«"init of function create main config"«std::endl;
00489     codeconfig = (create_code(create_row(line))&MASK_CODE);
00490     //std::cout«"end of function create main config"«std::endl;
00491     return(codeconfig);
00492 }
00493
00494 static void read_header_file(std::fstream &osheet, std::string &line, type_::UINT64 size)
00495 {
00496
00497     for(type_::UINT64 ii = 0; ii < 2+size; ii++)
00498         getline(osheet, line);
00499 }
00500
00501
00502 type_::UINT64 config::configimpl::create_T4code(type_::UINT64 main_code)
00503 {
00504  type_::UINT64 lcode = 0x803U;
00505  if(checkmaincode(main_code) == type_::TRUE)
00506  {
00507         // check the board in old Teloc in order to set new boards
00508      lcode |= teloc4000impl_::setSABO(main_code);
00509      lcode |= teloc4000impl_::setTECA(main_code);
00510      lcode |= teloc4000impl_::setDIGITAL(main_code);
00511      lcode |= teloc4000impl_::setBUS(main_code);
00512      lcode |= teloc4000impl_::setGPS(main_code);
00513      lcode |= teloc4000impl_::setCPM(main_code);
00514  }
00515  return(lcode);
00516 }
00517
00518 type_::UINT64 config::configimpl::create_T3code(type_::UINT64 main_code)
00519 {
00520     type_::UINT64 lcode = 0x1803U;
00521     if(checkmaincode(main_code) == type_::TRUE)
00522     {
00523         lcode |= teloc3000impl_::setSABO(main_code);
00524         lcode |= teloc3000impl_::setTACA(main_code);
00525         lcode |= teloc3000impl_::setDIGITAL(main_code);
00526         lcode |= teloc3000impl_::setBUS(main_code);
00527         lcode |= teloc3000impl_::setCPM(main_code);
00528
00529     }
00530     return(lcode);
00531 }
00532
00533
00534
00535
00536
00537 void config::configimpl::compare_create_configuration(std::fstream &osheet)
00538 {
00539
00540     std::string line;
00541     const std::string separator = ";";
00542     type_::UINT64 main_code = 0x0U;
00543     type_::UINT64 code = 0x0U;
00544     type_::UINT64 match = 0x0U;
00545     type_::UINT64 jj = 0;
00546     std::string assembly_code;
00547     std::string customer;
00548     std::vector<string> row;
00549     fCompare.open ("TelocMatch_"+gettelocstruct()->kindofTeloc+".csv",ios::app);
00550     fCompare« "AssemblyCode "«separator;
00551     fCompare«"Customer "«separator;
00552     if(gettelocstruct()->kindofTeloc == "T2500")
00553         fCompare« " Teloc2500 code  " «separator;
00554     else fCompare« " Teloc1500 code  " «separator;
00555     fCompare« "number of match"«separator;
00556     fCompare« "Teloc4000 code"«separator;
00557     fCompare« "Teloc3000 code"«endl;
00558
```

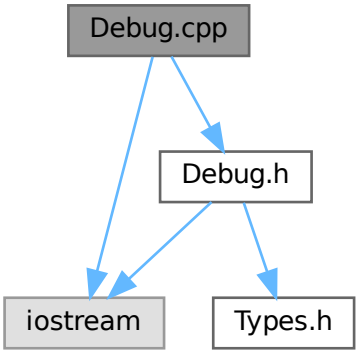
```
00559     std::cout«"compare_create_configuration"«std::endl;
00560
00561     if(osheet.is_open())
00562     {
00563         //read first row, and set configuration teloc under test!!!
00564         for(type_::UINT64 jj = 0; jj < getconfigstruct()->index_row-1; jj++)
00565         {
00566             //std::cout«"index_row = "«getconfigstruct()->index_row«std::endl;
00567         read_header_file(osheet, line, jj+1);
00568         assembly_code = line.substr(0, 13);
00569         //std::cout«"assembly_code = "«assembly_code«std::endl,
00570         row = create_row(line);
00571         //std::cout«"row assembly = "«row.at(0)«std::endl;
00572         customer = row.at(1);
00573         //std::cout«"customer = "«customer«std::endl;
00574         main_code = create_main_config(line);
00575         //std::cout«"main_code = "«main_code«std::endl;
00576         osheet.clear();
00577         osheet.seekg (0, ios::beg);
00578         read_header_file(osheet, line, 0);
00579         //std::cout«"main_code = "«main_code«std::endl;
00580         while(!osheet.eof())
00581         {
00582             getline(osheet, line);
00583             //std::cout«" while line = "«line«std::endl;
00584             if(line != "")
00585             {
00586                 code = create_code(create_row(line));
00587                 //std::cout«"code = "«code«std::endl;
00588             }//if
00589             //compare the code
00590             if (compare_handle(main_code, code) == type_::TRUE)
00591                 match++;
00592             //std::cout«"number match = "«match«std::endl;
00593         }//While
00594         osheet.clear();
00595         osheet.seekg (0, ios::beg);
00596         //read_header_file(osheet, line, jj+1);
00597         //osheet«line;
00598         //osheet«match«endl;
00599         {
00600             fCompare«assembly_code«separator,
00601             fCompare«customer«separator;
00602             fCompare«main_code«separator;
00603             fCompare«match-1«separator;
00604             fCompare«create_T4code(main_code)«separator;
00605             fCompare«create_T3code(main_code)«endl;
00606             match = 0x0U;
00607         }
00608     }//if
00609         //osheet.seekg(0, osheet.beg);
00610     }//for
00611     osheet.close();
00612     fCompare.close();
00613 }
00614
```

6.57 Debug.cpp File Reference

In this file are implemented all methods used for debug.

```
#include <iostream>
#include "Debug.h"
```

Include dependency graph for Debug.cpp:



6.57.1 Detailed Description

In this file are implemented all methods used for debug.

Author

Salvatore Muoio

Definition in file Debug.cpp.

6.58 Debug.cpp

Go to the documentation of this file.

```
00001
00007 #include <iostream>
00008 #include "Debug.h"
00009
00010
00011 type_::UINT8 debug_::enable = 1;
00012
00013
00014 void debug_::dummy(void)
00015 {
00016     //DEBUG_DISPLAY(debug_::enable, FUNCTION_NAME);
00017 }
00018
```

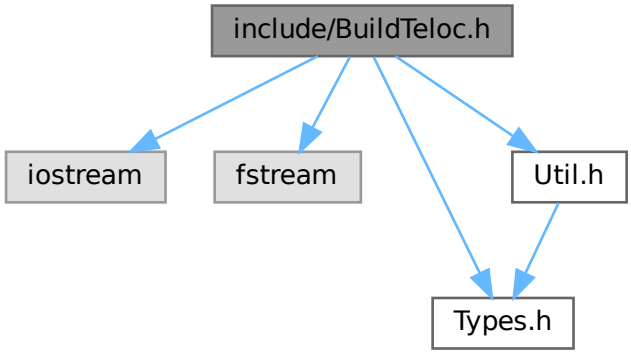
6.59 include/BuildTeloc.h File Reference

the file implements all methods used in config impl to build a teloc

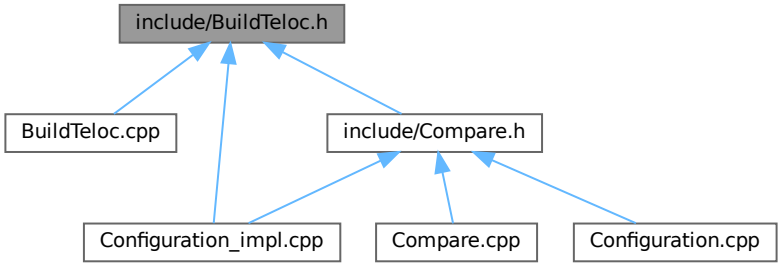
```
#include <iostream>
#include <fstream>
```

```
#include "Types.h"
#include "Util.h"
```

Include dependency graph for BuildTeloc.h:



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



Classes

- struct buildteloc_::t_buildtelocstruct

  generic parameter for a Teloc board
- struct buildteloc_::t_teloc_config

Namespaces

- namespace buildteloc_

Macros

- #define TELOC_BOARD 20

  maximum boards of Teloc
- #define DATABASE_FAMILY_TX500_SIZE 57
- #define DATABASE_FAMILY_TX500
- #define POSITION_TO_WRITING_SIZE 14

  size POSITION_TO_WRITING_SIZE
- #define POSITION_TO_WRITING

  database POSITION_TO_WRITING
- #define DATABASE_BOARD_T1500

  database of Teloc 1500
- #define DATABASE_BOARD_T2500

  database of Teloc 2500

- buildteloc_::t_buildtelocstruct buildtelocstruct [TELOC_BOARD][TELOC_BOARD]
- void lookuptableTeloc1500 (const type_::CHAR ∗const s, type_::UINT64 index, std::ofstream &myfile)

  the function build the teloc 1500
- void lookuptableTeloc2500 (const type_::CHAR ∗const s, type_::UINT64 index, std::ofstream &myfile)

  the function build the teloc 2500
- type_::ebool lookuptablefamily (const std::string family, std::string &name)

  !!!
- type_::UINT64 lookuptableposition (const std::string name)

6.59.1 Detailed Description

the file implements all methods used in config impl to build a teloc

Author

Salvatore Muoio

Definition in file BuildTeloc.h.

6.59.2 Macro Definition Documentation

6.59.2.1 DATABASE_BOARD_T1500

```
#define DATABASE_BOARD_T1500
```

Value:

```
{{"5.2420.360/01","POSU"},\
{"5.2420.361/01","POSU"},\
{"5.2420.205/07","IOCO"},\
{"5.2420.206/02","IOCO"},\
{"5.2420.201/02","CORE"},\
{"5.2621.212/02","DAIO"},\
{"5.2621.212/05","DAIO"},\
{"5.2621.212/12","DAIO"},\
{"5.2621.226/08","REBO"},\
{"5.2621.347/02","MVB"},\
{"5.2621.347/05","MVB"},\
{"5.2621.052/64","CPM" },\
{"5.2420.315/04","SRAM"},\
{"5.5005.200/01","SABOA"},\
{"5.2420.320/01","BAPLB"},\
{"5.2420.367/01","BAPLI"},\
{"5.2420.310/05","FLASH"},\
}\
```

database of Teloc 1500

Definition at line 115 of file BuildTeloc.h.

6.59.2.2 DATABASE_BOARD_T2500

```
#define DATABASE_BOARD_T2500
```

Value:

```
{{"5.2420.361/01", "POSU"},\
{"5.2621.212/02", "DAIO"},\
{"5.2621.202/03", "MAINC"},\
{"5.2621.306/01" ,"BUPLB"},\
{"5.2621.226/08", "REBO" },\
{"5.5005.200/03", "SABOA"},\
{"5.2621.335/01", "PC104"},\
{"5.2621.050/32", "CPM" },\
}\
```

database of Teloc 2500

Definition at line 138 of file BuildTeloc.h.

6.59.2.3 DATABASE_FAMILY_TX500

```
#define DATABASE_FAMILY_TX500
```

Definition at line 27 of file BuildTeloc.h.

6.59.2.4 DATABASE_FAMILY_TX500_SIZE

```
#define DATABASE_FAMILY_TX500_SIZE 57
```

Definition at line 22 of file BuildTeloc.h.

6.59.2.5 POSITION_TO_WRITING

```
#define POSITION_TO_WRITING
```

Value:

```
{{"POSU", "3"},\
{"CORE", "4"},\
{"IOCO", "5"},\
{"DAIO", "6"},\
{"REBO", "7"},\
{"SABO", "8"},\
{"MVB", "9"}, \
{"CAN", "10"}, \
{"GPS", "11"}, \
{"CPM", "12"},\
{"SRAM", "13"},\
{"FLASH", "14"},\
{"BACKPLANE", "15"},\
{"DATRA", "16"},\
}\
```

database POSITION_TO_WRITING

Definition at line 95 of file BuildTeloc.h.

6.59.2.6 POSITION_TO_WRITING_SIZE

```
#define POSITION_TO_WRITING_SIZE 14
```

size POSITION_TO_WRITING_SIZE

Definition at line 90 of file BuildTeloc.h.

6.59.2.7 TELOC_BOARD

```
#define TELOC_BOARD 20
```

maximum boards of Teloc

Definition at line 17 of file BuildTeloc.h.

6.59.3 Function Documentation

6.59.3.1 lookuptablefamily()

```
type_::ebool lookuptablefamily (
            const std::string family,
            std::string & name )
```

!!!

the function check the family code for the boards

Parameters

family,value	to check in the database
name,the	board name linked to the family code

Returns

board name

Definition at line 142 of file BuildTeloc.cpp.

6.59.3.2 lookuptableposition()

```
type_::UINT64 lookuptableposition (
            const std::string name )
```

Definition at line 162 of file BuildTeloc.cpp.

6.59.3.3 lookuptableTeloc1500()

```
void lookuptableTeloc1500 (
            const type_::CHAR *const s,
            type_::UINT64 index,
            std::ofstream & myfile )
```

the function build the teloc 1500

Parameters

∗s,pointer	to be string read in xlsx file

Returns

Definition at line 66 of file BuildTeloc.cpp.

6.59.3.4 lookuptableTeloc2500()

```
void lookuptableTeloc2500 (
            const type_::CHAR *const s,
            type_::UINT64 index,
            std::ofstream & myfile )
```

the function build the teloc 2500

Parameters

∗s,pointer	to be string read in xlsx file

Returns

Definition at line 104 of file BuildTeloc.cpp.

6.59.4 Variable Documentation

6.59.4.1 buildtelocstruct

```
buildteloc_::t_buildtelocstruct buildtelocstruct[TELOC_BOARD][TELOC_BOARD]  [extern]
```

Definition at line 19 of file BuildTeloc.cpp.

6.60 BuildTeloc.h

Go to the documentation of this file.

```
00001
00007 #ifndef __BUILD_TELOC_H__
00008 #define __BUILD_TELOC_H__
00009 #include <iostream>
00010 #include <fstream>
00011 #include "Types.h"
00012 #include "Util.h"
00017 #define TELOC_BOARD 20
00022 #define DATABASE_FAMILY_TX500_SIZE 57
00027 #define DATABASE_FAMILY_TX500 {{"1.341", "POSU"},\
00028                                {"0.360", "POSU"},\
00029                                {"0.361", "POSU"},\
00030                                {"0.260", "POSU"},\
00031                                {"0.361", "POSU"},\
00032                                {"0.362", "POSU"},\
00033                                {"0.200", "CORE"},\
00034                                {"0.201", "CORE"},\
00035                                {"1.201", "CORE"},\
00036                                {"0.300", "CORE"},\
00037                                {"0.301", "CORE"},\
00038                                {"1.301", "CORE"},\
00039                                {"1.202", "CORE"},\
00040                                {"1.302", "CORE"},\
00041                                {"1.212", "DAIO"},\
00042                                {"0.312", "DAIO"},\
00043                                {"1.312", "DAIO"},\
00044                                {"1.226", "REBO"},\
00045                                {"1.326", "REBO"},\
00046                                {"0.205", "IOCO"},\
00047                                {"0.206", "IOCO"},\
00048                                {"0.306", "IOCO"},\
00049                                {"1.235", "CAN"},\
00050                                {"5.200", "SABO"},\
00051                                {"5.201", "SABO"},\
00052                                {"5.300", "SABO"},\
00053                                {"5.301", "SABO"},\
00054                                {"1.247", "MVB"}, \
00055                                {"1.347", "MVB"}, \
00056                                {"1.348", "MVB"}, \
00057                                {"1.235", "CAN"}, \
00058                                {"1.335", "CAN"}, \
00059                                {"0.230", "GPS"}, \
00060                                {"0.330", "GPS"}, \
00061                                {"0.315", "SRAM"},\
00062                                {"0.310", "FLASH"},\
00063                                {"0.320", "BACKPLANE"},\
00064                                {"0.321", "BACKPLANE"},\
00065                                {"0.322", "BACKPLANE"},\
00066                                {"0.323", "BACKPLANE"},\
00067                                {"0.324", "BACKPLANE"},\
00068                                {"0.365", "BACKPLANE"},\
00069                                {"0.366", "BACKPLANE"},\
00070                                {"0.367", "BACKPLANE"},\
00071                                {"1.306", "BACKPLANE"},\
00072                                {"1.050", "CPM"},\
00073                                {"1.052", "CPM"},\
00074                                {"0.548", "CPM"},\
00075                                {"1.137", "CPM"},\
00076                                {"1.053", "CPM"},\
00077                                {"1.159", "CPM"},\
00078                                {"1.160", "CPM"},\
00079                                {"0.566", "CPM"},\
00080                                {"0.138", "CPM"},\
00081                                {"0.548", "CPM"},\
00082                                {"1.271", "DATRA"},\
00083                                {"1.371", "DATRA"},\
00084     }\
00085
00090 #define POSITION_TO_WRITING_SIZE 14
00095 #define POSITION_TO_WRITING {{"POSU", "3"},\
00096                              {"CORE", "4"},\
00097                              {"IOCO", "5"},\
00098                              {"DAIO", "6"},\
00099                              {"REBO", "7"},\
00100                              {"SABO", "8"},\
00101                              {"MVB", "9"}, \
00102                              {"CAN", "10"}, \
00103                              {"GPS", "11"}, \
00104                              {"CPM", "12"},\
00105                              {"SRAM", "13"},\
00106                              {"FLASH", "14"},\
00107                              {"BACKPLANE", "15"},\
```

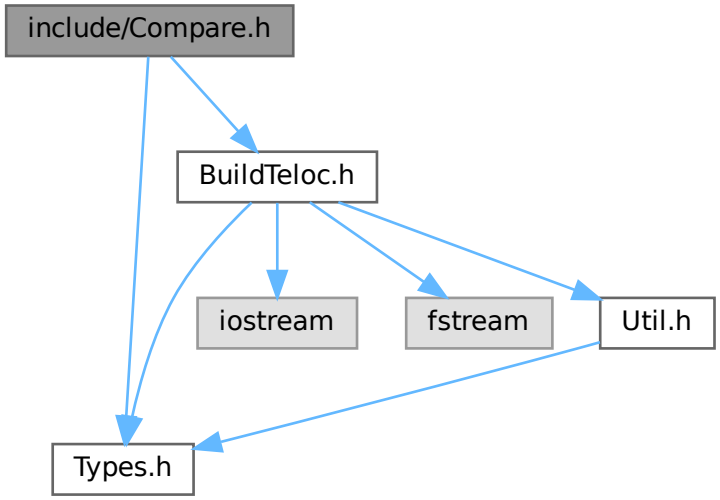
```
00108                               {"DATRA", "16"},\
00109                               }\
00110
00115 #define DATABASE_BOARD_T1500 {{"5.2420.360/01","POSU"},\
00116                                {"5.2420.361/01","POSU"},\
00117                                {"5.2420.205/07","IOCO"},\
00118                                {"5.2420.206/02","IOCO"},\
00119                                {"5.2420.201/02","CORE"},\
00120                                {"5.2621.212/02","DAIO"},\
00121                                {"5.2621.212/05","DAIO"},\
00122                                {"5.2621.212/12","DAIO"},\
00123                                {"5.2621.226/08","REBO"},\
00124                                {"5.2621.347/02","MVB"},\
00125                                {"5.2621.347/05","MVB"},\
00126                                {"5.2621.052/64","CPM" },\
00127                                {"5.2420.315/04","SRAM"},\
00128                                {"5.5005.200/01","SABOA"},\
00129                                {"5.2420.320/01","BAPLB"},\
00130                                {"5.2420.367/01","BAPLI"},\
00131                                {"5.2420.310/05","FLASH"},\
00132                               }\
00133
00138 #define DATABASE_BOARD_T2500 {{"5.2420.361/01", "POSU"},\
00139                                {"5.2621.212/02", "DAIO"},\
00140                                {"5.2621.202/03", "MAINC"},\
00141                                {"5.2621.306/01" ,"BUPLB"},\
00142                                {"5.2621.226/08", "REBO" },\
00143                                {"5.5005.200/03", "SABOA"},\
00144                                {"5.2621.335/01", "PC104"},\
00145                                {"5.2621.050/32", "CPM" },\
00146                                }\
00147
00148
00149 // namespace::buildteloc_
00151 namespace buildteloc_
00152 {
00157     typedef struct
00158     {
00159         std::string board_name;
00160         type_::ebool active;
00161         type_::UINT64 numberofboard;
00162     }t_buildtelocstruct;
00163     typedef struct
00164     {
00165         type_::UINT64 matchvalue;
00166         type_::UINT64 posu;
00167         type_::UINT64 core;
00168         type_::UINT64 ioco;
00169         type_::UINT64 daio;
00170         type_::UINT64 rebo;
00171         type_::UINT64 sabo;
00172         type_::UINT64 mvb;
00173         type_::UINT64 can;
00174         type_::UINT64 gps;
00175         type_::UINT64 cpm;
00176         type_::UINT64 sram;
00177         type_::UINT64 flash;
00178         type_::UINT64 backplane;
00179     }t_teloc_config;
00180 }//namespace
00181
00182 extern buildteloc_::t_buildtelocstruct buildtelocstruct[TELOC_BOARD][TELOC_BOARD];
00184
00190 void lookuptableTeloc1500(const type_::CHAR * const s, type_::UINT64 index, std::ofstream &myfile);
00197 void lookuptableTeloc2500(const type_::CHAR * const s, type_::UINT64 index, std::ofstream &myfile);
00205 type_::ebool lookuptablefamily(const std::string family, std::string &name);
00213 type_::UINT64 lookuptableposition(const std::string name);
00214
00215
00216
00217 #endif
```

6.61 include/Compare.h File Reference

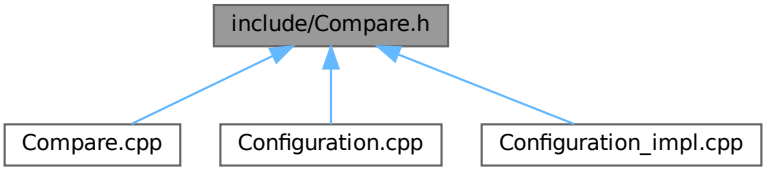
in this file are implmented the methods used to comapre the dirrent telocs read

```
#include "Types.h"
#include "BuildTeloc.h"
```

Include dependency graph for Compare.h:



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



Namespaces

- namespace compare_

Macros

- #define TABLE_MATCH_VALUE

  the table manages the difference of board in the checking

Functions

- type_::UINT64 plausibilitycheck_numberboard (buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[ ][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
- type_::UINT64 plausibilitycheck_boards (buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[ ][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj)
- type_::ebool compare_handle (type_::UINT64 main_config, type_::UINT64 code_config)

  the function manages the main compare

6.61.1 Detailed Description

in this file are implmented the methods used to comapre the dirrent telocs read

Author

Salvatore Muoio

Definition in file Compare.h.

6.61.2 Macro Definition Documentation

6.61.2.1 TABLE_MATCH_VALUE

```
#define TABLE_MATCH_VALUE
```

Value:

```
{{"POSU","10"},\
{"IOCO","10"},\
{"CORE","10"},\
{"DAIO","10"},\
{"REBO","10"},\
{"MVB","10"},\
{"CPM","10"},\
{"SRAM","10"},\
{"SABOA","10"},\
{"BAPLB","10"},\
{"BAPLI","10"},\
{"FLASH","10"},\
}\
```

the table manages the difference of board in the checking

Definition at line 15 of file Compare.h.

6.61.3 Function Documentation

6.61.3.1 compare_handle()

```
type_::ebool compare_handle (
            type_::UINT64 main_config,
            type_::UINT64 code_config )
```

the function manages the main compare

Parameters

main_config,main	configuration
code_config	configuration code

Returns

value of not match

Definition at line 119 of file Compare.cpp.

6.61.3.2 plausibilitycheck_boards()

```
type_::UINT64 plausibilitycheck_boards (
             buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[][TELOC_BOARD],
             type_::UINT64 len,
             type_::UINT64 jj )
```

Definition at line 88 of file Compare.cpp.

6.61.3.3 plausibilitycheck_numberboard()

```
type_::UINT64 plausibilitycheck_numberboard (
             buildteloc_::t_buildtelocstruct ptrbuildtelocstruc[][TELOC_BOARD],
             type_::UINT64 len,
             type_::UINT64 jj )
```

Definition at line 61 of file Compare.cpp.

6.62 Compare.h

Go to the documentation of this file.

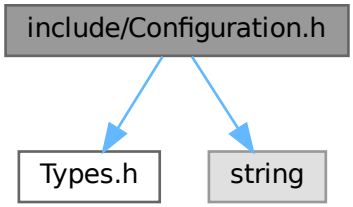
```
00001
00006 #ifndef __COMPARE_H__
00007 #define __COMPARE_H__
00008 #include "Types.h"
00009 #include "BuildTeloc.h"
00010
00015 #define TABLE_MATCH_VALUE {{"POSU","10"},\
00016                            {"IOCO","10"},\
00017                            {"CORE","10"},\
00018                            {"DAIO","10"},\
00019                            {"REBO","10"},\
00020                            {"MVB","10"},\
00021                            {"CPM","10"},\
00022                            {"SRAM","10"},\
00023                            {"SABOA","10"},\
00024                            {"BAPLB","10"},\
00025                            {"BAPLI","10"},\
00026                            {"FLASH","10"},\
00027                            }\
00028
00029
00030
00031
00032 namespace compare_
00033 {
00034     // create a structure for the Compare module
00035
00036
00037 }
00038
00047 type_::UINT64 plausibilitycheck_numberboard(buildteloc_::t_buildtelocstruct
      ptrbuildtelocstruc[][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj);
00048
00057 type_::UINT64 plausibilitycheck_boards(buildteloc_::t_buildtelocstruct
      ptrbuildtelocstruc[][TELOC_BOARD], type_::UINT64 len, type_::UINT64 jj);
00065 type_::ebool compare_handle(type_::UINT64 main_config, type_::UINT64 code_config);
00066
00067
00068 #endif
```

6.63 include/Configuration.h File Reference

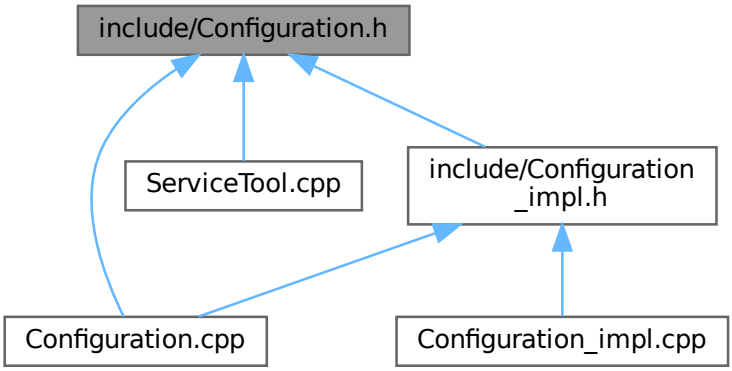
in this file are implemented the methods used to work with Configuration file

```
#include "Types.h"
#include <string>
```

Include dependency graph for Configuration.h:



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



Classes

- struct config_::t_configstruct
- struct config_::t_telocstrcut
- class config

  the class manages the methods to parser the configuration file

Namespaces

- namespace config_

  the namespace by Configuration file

Macros

- #define max_size_tab 10000

6.63.1 Detailed Description

in this file are implemented the methods used to work with Configuration file

Author

Salvatore Muoio

Definition in file Configuration.h.

6.63.2 Macro Definition Documentation

6.63.2.1 max_size_tab

```
#define max_size_tab 10000
```

Definition at line 12 of file Configuration.h.

6.64 Configuration.h

Go to the documentation of this file.

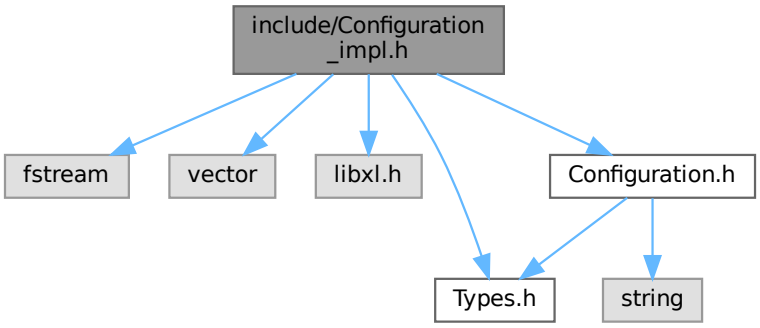
```
00001
00006 #ifndef __CONFIGURATION_H__
00007 #define __CONFIGURATION_H__
00008
00009 #include"Types.h"
00010 #include <string>
00011
00012 #define max_size_tab 10000
00017 namespace config_
00018 {
00019    typedef struct
00020     {
00021         type_::UINT64 index_row;
00022         std::string line;
00023         std::string title;
00024         type_::CHAR filename[20];
00025         type_::CHAR column[60];
00026         type_::UINT64 numberboardTeloc[20];
00027         type_:: CHAR assemblycode[20][20];
00028         //char *kindofTeloc;
00029     }t_configstruct;
00030
00035     typedef struct
00036     {
00037         std::string kindofTeloc;
00038         type_::UINT64 Teloc;
00039     }t_telocstrcut;
00040 }
00041
00046 class config
00047 {
00048     protected:
00049     static config_::t_configstruct *getconfigstruct(void);
00050     static config_::t_telocstrcut *gettelocstruct(void);
00051     private:
00052     class configimpl;
00053     configimpl *pimpl;
00054     config();
00055     virtual ~config(){};
00056     config(const config &);
00057     const config &operator = (const config &);
00058     std::string whoamI(std::string line);
00059     public:
00060     static config &getinstance();
00061     void readfileconfig(void);
00062 };
00063
00064
00065 #endif
```

6.65 include/Configuration_impl.h File Reference

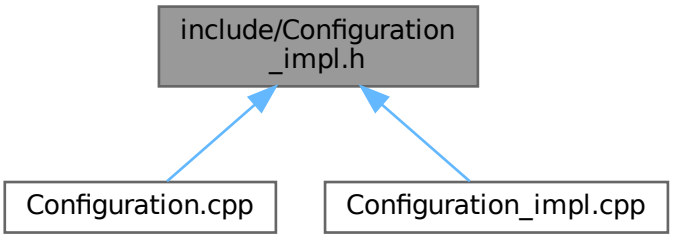
in this file are implemented the methods private in the pimpl

```
#include <fstream>
#include <vector>
#include "libxl.h"
#include "Configuration.h"
#include "Types.h"
```

Include dependency graph for Configuration_impl.h:



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



Classes

- struct configimpl_::t_configstructimpl
- struct configimpl_::t_filestruct

  the scructure is the file row to fullfill
- struct config::configimpl

Namespaces

- namespace configimpl_

  the namespace used from the pimpl

Macros

- #define MASK_CODE 0x3FFF

  mask code used to create the code config
- #define FAMILY_TELOC_1500_SIZE 14

  size del database TELOC 1500
- #define FAMILY_TELOC_1500 {"POSU", "CORE", "IOCO", "DAIO", "REBO", "SABO", "MVB", "CAN", "GPS", "CPM", "SRAM", "FLASH", "BACKPLANE", "DATRA"};
- #define FAMILY_TELOC_2500_SIZE 14
- #define FAMILY_TELOC_2500 {"POSU", "CORE", "", "DAIO", "REBO", "SABO", "MVB", "", "GPS", "CPM", "SRAM", "FLASH", "BACKPLANE", "DATRA"};

Functions

- t_configstructimpl * configimpl_::getaccescfgimpl (void)
- t_filestruct * configimpl_::getfilestruct (void)
- void closefile (void)

6.65.1 Detailed Description

in this file are implemented the methods private in the pimpl

Author

Salvatore Muoio

Definition in file Configuration_impl.h.

6.65.2 Macro Definition Documentation

6.65.2.1 FAMILY_TELOC_1500

```
#define FAMILY_TELOC_1500 {"POSU", "CORE", "IOCO", "DAIO", "REBO", "SABO", "MVB", "CAN", "GPS",
"CPM", "SRAM", "FLASH", "BACKPLANE", "DATRA"};
```

Definition at line 31 of file Configuration_impl.h.

6.65.2.2 FAMILY_TELOC_1500_SIZE

```
#define FAMILY_TELOC_1500_SIZE 14
```

size del database TELOC 1500

Definition at line 26 of file Configuration_impl.h.

6.65.2.3 FAMILY_TELOC_2500

```
#define FAMILY_TELOC_2500 {"POSU", "CORE", "", "DAIO", "REBO", "SABO", "MVB", "", "GPS", "CPM",
"SRAM", "FLASH", "BACKPLANE", "DATRA"};
```

Definition at line 41 of file Configuration_impl.h.

6.65.2.4 FAMILY_TELOC_2500_SIZE

```
#define FAMILY_TELOC_2500_SIZE 14
```

Definition at line 36 of file Configuration_impl.h.

6.65.2.5 MASK_CODE

```
#define MASK_CODE 0x3FFF
```

mask code used to create the code config

Definition at line 21 of file Configuration_impl.h.

6.65.3 Function Documentation

6.65.3.1 closefile()

```
void closefile (
            void  )
```

Definition at line 46 of file Configuration_impl.cpp.

6.66 Configuration_impl.h

Go to the documentation of this file.

```
00001
00007 #ifndef __CONFIGURATION_IMPL_H__
00008 #define __CONFIGURATION_IMPL_H__
00009
00010 #include <fstream>
00011 #include <vector>
00012 #include "libxl.h"
00013 #include "Configuration.h"
00014 #include "Types.h"
00015
00016 using namespace libxl;
00021 #define MASK_CODE 0x3FFF
00026 #define FAMILY_TELOC_1500_SIZE 14
00031 #define FAMILY_TELOC_1500 {"POSU", "CORE", "IOCO", "DAIO", "REBO", "SABO", "MVB", "CAN",  "GPS",
      "CPM", "SRAM", "FLASH", "BACKPLANE", "DATRA"};
00036 #define FAMILY_TELOC_2500_SIZE 14
00041 #define FAMILY_TELOC_2500 {"POSU", "CORE", "", "DAIO", "REBO", "SABO", "MVB", "", "GPS", "CPM",
      "SRAM", "FLASH", "BACKPLANE", "DATRA"};
00042
00047 namespace configimpl_
00048 {
00049     typedef struct
00050     {
00051         type_::UINT64 findcolumn;
00052     }t_configstructimpl;
```

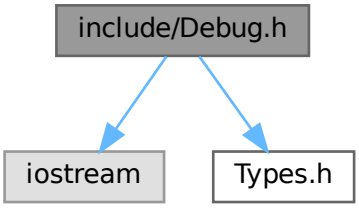
```
00053
00058     typedef struct
00059     {
00060         std::string assembly_code;
00061         std::string customer;
00062         std::string posu;
00063         std::string core;
00064         std::string ioco;
00065         std::string daio;
00066         std::string rebo;
00067         std::string sabo;
00068         std::string mvb;
00069         std::string can;
00070         std::string gps;
00071         std::string cpm;
00072         std::string sram;
00073         std::string flash;
00074         std::string backplane;
00075         std::string datra;
00076     }t_filestruct;
00077
00078     extern t_configstructimpl *getaccescfgimpl(void);
00079     extern t_filestruct *getfilestruct(void);
00080 }
00081
00082 void closefile(void);
00083
00088 struct config::configimpl
00089 {
00090     configimpl(){};
00091     virtual ~configimpl(){
00092     };
00093     void extract_filename(std::string line, type_::CHAR *filename, type_::CHAR* assemblycode);
00094     void extract_column_compare(std::string line, type_::CHAR *col);
00095     type_::ebool find_column(const type_::CHAR *title, const type_::CHAR *col);
00096     type_::ebool parser_kenfile(const type_:: CHAR *col, std::string filename);
00097     type_::UINT64 getsizeTeloc(void);
00098     type_::UINT64 create_T4code(type_::UINT64 main_code);
00099     type_::UINT64 create_T3code(type_::UINT64 main_code);
00100     void scroll_column(const std::string teloccode);
00101     void create_template(ofstream &osheet, std::string teloc);
00102     void create_teloc_assembly(const char *s, Sheet *osheet,  type_::UINT64 row);
00103     void create_output_file(std::vector<std::string> col, ofstream &file);
00104     void extract_family(std::string code, ofstream &osheet, configimpl_::t_filestruct *ptr);
00105     void extract_version(std::string code, std::string &variant);
00106     void write_variant(type_::UINT64 pos, std::string variant, configimpl_::t_filestruct *ptr);
00107     void compare_create_configuration(fstream &osheet);
00108 };
00109
00110
00111 #endif
```

6.67 include/Debug.h File Reference

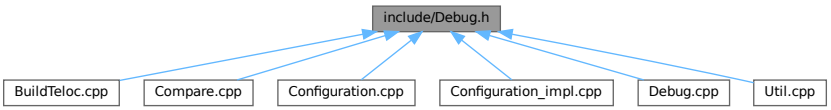
In this file are implemented all methods used for debug.

```
#include <iostream>
#include "Types.h"
```

Include dependency graph for Debug.h:



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



Classes

- class debug

Namespaces

- namespace debug_

  namespace used to manages the debug

Macros

- #define DEBUG_DISPLAY(enable, exp) if(enable) debug_::dbg_display(exp)

  macro function used to call the template function
- #define DEBUG_ENABLE(exp) if(exp)
- #define FUNCTION_NAME __func__

Functions

- void debug_::dbg_info (void)
- template<typename T >
  T debug_::dbg_display (T &x)
- void debug_::dummy (void)

Variables

- type_::UINT8 debug_::enable = 1

  variable used to enable the function to display the info

6.67.1 Detailed Description

In this file are implemented all methods used for debug.

Author

Salvatore Muoio

Definition in file Debug.h.

6.67.2 Macro Definition Documentation

6.67.2.1 DEBUG_DISPLAY

```
#define DEBUG_DISPLAY(
              enable,
              exp ) if(enable) debug_::dbg_display(exp)
```

macro function used to call the template function

Definition at line 20 of file Debug.h.

6.67.2.2 DEBUG_ENABLE

```
#define DEBUG_ENABLE(
              exp ) if(exp)
```

Definition at line 21 of file Debug.h.

6.67.2.3 FUNCTION_NAME

```
#define FUNCTION_NAME __func__
```

Definition at line 22 of file Debug.h.

6.68 Debug.h

Go to the documentation of this file.

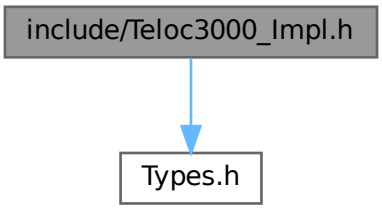
```
00001
00007 #ifndef __DEBUG_H__
00008 #define __DEBUG_H__
00009
00010 #include<iostream>
00011 #include "Types.h"
00012
00013 using namespace std;
00014
00015 //#define DEBUG_INFO(exp) ((exp) ? debug_::dbg_info() : debug_::dummy())
00020 #define DEBUG_DISPLAY(enable, exp) if(enable) debug_::dbg_display(exp)
00021 #define DEBUG_ENABLE(exp) if(exp)
00022 #define FUNCTION_NAME __func__
00023
00028 namespace debug_
00029 {
00034     extern type_::UINT8 enable;
00035     void dbg_info(void);
00040     template <typename T> T dbg_display(T &x ){
00041      std::cout«x«std::endl;
00042      return 0;
00043     }
00044     void dummy(void);
00045 }
00046
00047 class debug
00048 {
00049     public:
00050     debug();
00051     void debuginfo(void);
00052     virtual ~debug(){};
00053
00054 };
00055
00056 #endif
```

6.69 include/Teloc3000_Impl.h File Reference

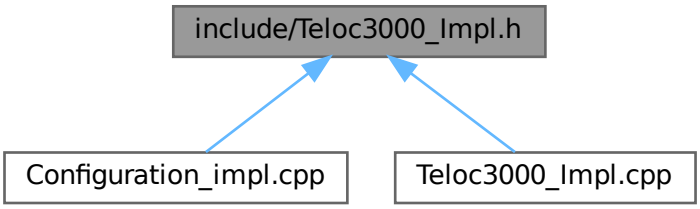
file used to manage the methods for Teloc3000

```
#include "Types.h"
```

Include dependency graph for Teloc3000_Impl.h:



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



Namespaces

- namespace teloc3000impl_

  *namspaxe teloc3000impl_*

Enumerations

- enum teloc3000impl_::eT3Code {
  teloc3000impl_::NO_BOARD_ENABLED = 0 , teloc3000impl_::DATRA_ENABLED = 0x04 , teloc3000impl_::CAN_ENABLED = 0x08 , teloc3000impl_::USCOA_ENABLED = 0x10 ,
  teloc3000impl_::DAIOD_ENABLED = 0x20 , teloc3000impl_::REBO_ENABLED = 0x40 , teloc3000impl_::TACHA_ENABLED = 0x80 , teloc3000impl_::USCOA_TACHA_ENABLED = 0x90 ,
  teloc3000impl_::SABO_ENABLED = 0x100 , teloc3000impl_::MVB_ENABLED = 0x200 , teloc3000impl_::CPM_ENABLED = 0x400 }

  the structure manage the boards for the Teloc 3000

Functions

- eT3Code teloc3000impl_::setSABO (type_::UINT64 maincode)

  the function set the SABO, if it's set on the old Teloc
- eT3Code teloc3000impl_::setDIGITAL (type_::UINT64 maincode)

  the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc
- eT3Code teloc3000impl_::setTACA (type_::UINT64 maincode)

  the function set the setTACA and the USCOA, if they are set on the old Teloc
- eT3Code teloc3000impl_::setBUS (type_::UINT64 maincode)

  the function set the MVB or CAN bus , if they are set on the old Teloc
- eT3Code teloc3000impl_::setCPM (type_::UINT64 maincode)

  the function set the CPM , if it's set on the old Teloc

6.69.1 Detailed Description

file used to manage the methods for Teloc3000

Author

: Salvatore Muoio

Definition in file Teloc3000_Impl.h.

6.70 Teloc3000_Impl.h

Go to the documentation of this file.

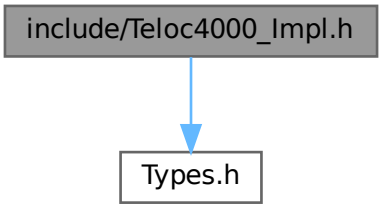
```
00001
00007 #ifndef INCLUDE_TELOC3000_IMPL_H_
00008 #define INCLUDE_TELOC3000_IMPL_H_
00009
00010
00011 #include "Types.h"
00012
00014 namespace teloc3000impl_
00015 {
00021     typedef enum
00022     {
00023         NO_BOARD_ENABLED   = 0,
00024         DATRA_ENABLED      = 0x04,
00025         CAN_ENABLED        = 0x08,
00026         USCOA_ENABLED      = 0x10,
00027         DAIOD_ENABLED      = 0x20,
00028         REBO_ENABLED       = 0x40,
00029         TACHA_ENABLED      = 0x80,
00030         USCOA_TACHA_ENABLED = 0x90,
00031         SABO_ENABLED       = 0x100,
00032         MVB_ENABLED        = 0x200,
00033         CPM_ENABLED        = 0x400,
00034     }eT3Code;
00041     eT3Code setSABO(type_::UINT64 maincode);
00048     eT3Code setDIGITAL(type_::UINT64 maincode);
00055     eT3Code setTACA(type_::UINT64 maincode);
00062     eT3Code setBUS(type_::UINT64 maincode);
00069     eT3Code setCPM(type_::UINT64 maincode);
00070
00071
00072 }//namespace teloc3000impl_
00073
00074
00075 #endif /* INCLUDE_TELOC3000_IMPL_H_ */
```

6.71 include/Teloc4000_Impl.h File Reference

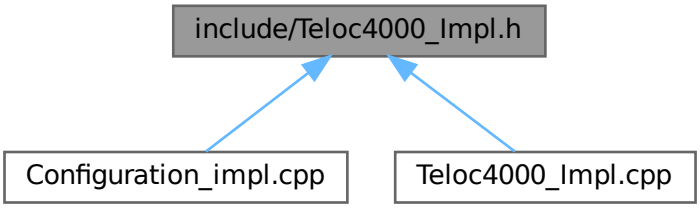
file used to manage the methods for Teloc4000

```
#include "Types.h"
```

Include dependency graph for Teloc4000_Impl.h:



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



Namespaces

- namespace teloc4000impl_

  *namespace teloc3000impl_*

Enumerations

- enum teloc4000impl_::eT4Code { teloc4000impl_::NO_BOARD_ENABLED = 0 , teloc4000impl_::ONLY_DRSCA_ENABLED = 0x08 , teloc4000impl_::TECA_DRSCA_ENABLED = 0x0C , teloc4000impl_::DAIOD_ENABLED = 0x10 , teloc4000impl_::DOCAA_ENABLED = 0x20 , teloc4000impl_::SABOC_ENABLED = 0x40 , teloc4000impl_::MVB_ENABLED = 0x80 , teloc4000impl_::CAN_ENABLED = 0x100 , teloc4000impl_::GPS_ENABLED = 0x200 , teloc4000impl_::CPM_ENABLED = 0x400 }

  the structure manage the board for the TEloc 4000

Functions

- eT4Code teloc4000impl_::setDIGITAL (type_::UINT64 maincode)

  the function set the DAIOD and DOCAA and the DRSCA, if they are set on the old Teloc
- eT4Code teloc4000impl_::setSABO (type_::UINT64 maincode)

  the function set the SABOC, if it's set on the old Teloc
- eT4Code teloc4000impl_::setTECA (type_::UINT64 maincode)

  the function set the setTECA and the DRSCA, if they are set on the old Teloc
- eT4Code teloc4000impl_::setBUS (type_::UINT64 maincode)

  the function set the MVB or CAN bus , if they are set on the old Teloc
- eT4Code teloc4000impl_::setGPS (type_::UINT64 maincode)

  the function set the GPS , if it's set on the old Teloc
- eT4Code teloc4000impl_::setCPM (type_::UINT64 maincode)

  the function set the CPM , if it's set on the old Teloc

6.71.1 Detailed Description

file used to manage the methods for Teloc4000

Author

: Salvatore Muoio

Definition in file Teloc4000_Impl.h.

6.72 Teloc4000_Impl.h

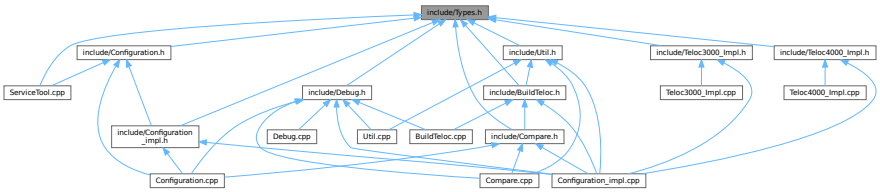
Go to the documentation of this file.

```
00001
00007 #include "Types.h"
00008
00010 namespace teloc4000impl_
00011 {
00017     typedef enum
00018     {
00019         NO_BOARD_ENABLED   = 0,
00020         ONLY_DRSCA_ENABLED = 0x08,
00021         TECA_DRSCA_ENABLED = 0x0C,
00022         DAIOD_ENABLED      = 0x10,
00023         DOCAA_ENABLED      = 0x20,
00024         SABOC_ENABLED      = 0x40,
00025         MVB_ENABLED        = 0x80,
00026         CAN_ENABLED        = 0x100,
00027         GPS_ENABLED        = 0x200,
00028         CPM_ENABLED        = 0x400,
00029     }eT4Code;
00036     eT4Code setDIGITAL(type_::UINT64 maincode);
00043     eT4Code setSABO(type_::UINT64 maincode);
00050     eT4Code setTECA(type_::UINT64 maincode);
00051     //teloc4000impl_::eT4Code setDIGITAL(type_::UINT64 maincode);
00058     eT4Code setBUS(type_::UINT64 maincode);
00065     eT4Code setGPS(type_::UINT64 maincode);
00072     eT4Code setCPM(type_::UINT64 maincode);
00073
00074 }
00075
00082 //type_::ebool checkmaincode(type_::UINT64 maincode);
```

6.73 include/Types.h File Reference

the file implements the custom typedef

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



Namespaces

- namespace type_

  namespace used to manages the typedef

Typedefs

- typedef unsigned short type_::UINT16
- typedef unsigned int type_::UINT64
- typedef unsigned char type_::UINT8
- typedef char type_::CHAR

Enumerations

- enum type_::ebool { type_::FALSE , type_::TRUE }
- enum type_::e_result { type_::RESULT_OK , type_::RESULT_POINTER_NOT_ADDRESSED , type_::RESULT_OUT_OF_RAN , type_::RESULT_NOT_READY_UART = 4 }

  the enum is used as validity check in the methods

6.73.1 Detailed Description

the file implements the custom typedef

Author

Salvatore Muoio

Definition in file Types.h.

6.74 Types.h

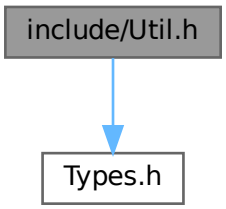
Go to the documentation of this file.

```
00001
00007 #ifndef __TYPE_H__
00008 #define __TYPE_H__
00009
00014 namespace type_
00015 {
00020     typedef unsigned short UINT16;
00021     typedef unsigned int UINT64;
00022     typedef unsigned char UINT8;
00023     typedef char CHAR;
00028     typedef enum
00029     {
00030         FALSE,
00031         TRUE,
00032     }ebool;
00033
00038     typedef enum
00039     {
00040         RESULT_OK,
00041         RESULT_POINTER_NOT_ADDRESSED,
00042         RESULT_OUT_OF_RANGE,
00043         RESULT_NOT_READY_UART = 4,
00044     }e_result;
00045 }
00046
00047
00048 #endif
```

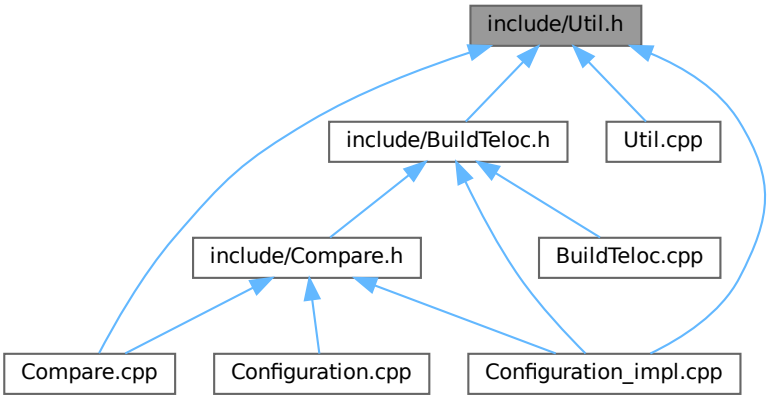
6.75 include/Util.h File Reference

In this file are implemented all methods util.

`#include "Types.h"`
Include dependency graph for Util.h:



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



Namespaces

- namespace util_

Functions

- type_::ebool util_::charpointer_compare (const type_::CHAR ∗a, const type_::CHAR ∗b)
- type_::e_result util_::CheckArg (void ∗pArg)

  the function checks if the pointer is addressed
- type_::UINT64 util_::ConverTelocCode2Num (std::string teloccode)

  the function returns which kinf of Teloc is under test

6.75.1 Detailed Description

In this file are implemented all methods util.

Author

Salvatore Muoio

Definition in file Util.h.

6.76 Util.h

Go to the documentation of this file.

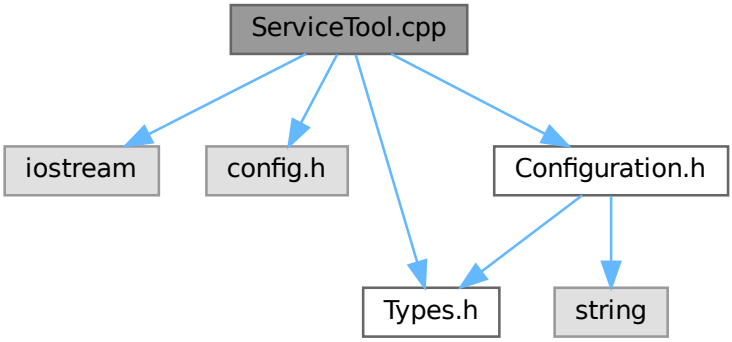
```
00001
00006 #ifndef __UTIL_H__
00007 #define __UTIL_H__
00008 #include "Types.h"
00009 // namespace::util_
00011 namespace util_
00012 {
00013     type_::ebool charpointer_compare(const type_::CHAR *a, const type_::CHAR *b);
00020     type_::e_result CheckArg(void *pArg);
00027     type_::UINT64 ConverTelocCode2Num(std::string teloccode);
00028
00029 }
00031
00032 #endif
```

6.77 ServiceTool.cpp File Reference

main of sw project

```
#include <iostream>
#include "config.h"
#include "Configuration.h"
#include "Types.h"
```

Include dependency graph for ServiceTool.cpp:



Classes

- class CodeT4< T >

Functions

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

6.77.1 Detailed Description

main of sw project

Author

Salvatore Muoio

Definition in file ServiceTool.cpp.

6.77.2 Function Documentation

6.77.2.1 main()

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

Definition at line 27 of file ServiceTool.cpp.

6.78 ServiceTool.cpp

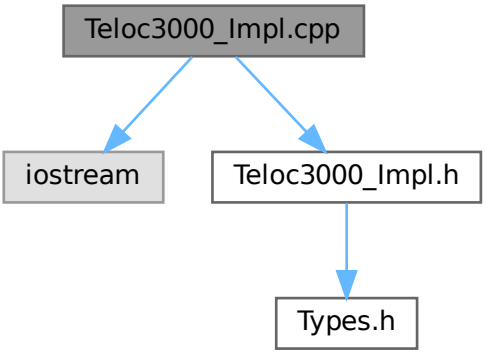
Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include "config.h"
00008 #include "Configuration.h"
00009 #include "Types.h"
00010 //#include "libxl.h"
00011
00012 using namespace std;
00013 //using namespace libxl;
00014
00015
00016 template <class T>
00017 class CodeT4
00018 {
00019     private:
00020         T T4Code;
00021     public:
00022         //T create_T4code(T main_code);
00023 };
00024
00025
00026
00027 int main(int argc, char **argv) {
00028 //  Book *book = xlCreateXMLBook();
00029 //  type_::ebool find = type_::FALSE;
00030     config &obj = config::getinstance();
00031     obj.readfileconfig();
00032     CodeT4<int> T4Code;
00033     std::cout « "Service Tool" « std::endl;
00034     std::cout « "Version " « ServiceTool_VERSION_MAJOR « "." « ServiceTool_VERSION_MINOR « std::endl;
00035     return 0;
00036 }
```

6.79 Teloc3000_Impl.cpp File Reference

```
#include "iostream"
#include "Teloc3000_Impl.h"
```

Include dependency graph for Teloc3000_Impl.cpp:



6.80 Teloc3000_Impl.cpp

Go to the documentation of this file.

```
00001
00008 #include "iostream"
00009 #include "Teloc3000_Impl.h"
00010
00011 using namespace teloc3000impl_;
00019 teloc3000impl_::eT3Code teloc3000impl_::setSABO(type_::UINT64 maincode)
00020 {
00021     constexpr type_::UINT64 MASK_SABO = 0x20U;
00022     type_::UINT64 lmain = maincode&MASK_SABO;
00023     return((lmain == MASK_SABO) ? (teloc3000impl_:: SABO_ENABLED) :
      (teloc3000impl_::NO_BOARD_ENABLED));
00024 }
00025
00033 teloc3000impl_::eT3Code teloc3000impl_::setTACA(type_::UINT64 maincode)
00034 {
00035     type_::UINT64 lret = teloc3000impl_:: NO_BOARD_ENABLED;
00036     constexpr type_::UINT64 MASK_IOCO = 0x04U;
00037     type_::UINT64 lmain = maincode&MASK_IOCO;
00038     if(lmain == MASK_IOCO)
00039     {
00040         if (setSABO(maincode) == teloc3000impl_:: SABO_ENABLED)
00041             lret = teloc3000impl_::USCOA_ENABLED;
00042         else
00043             lret = teloc3000impl_::USCOA_TACHA_ENABLED;
00044     }
00045     else
00046         lret = teloc3000impl_::NO_BOARD_ENABLED;
00047     return((teloc3000impl_::eT3Code)lret);
00048 }
00056 teloc3000impl_::eT3Code teloc3000impl_ :: setDIGITAL(type_::UINT64 maincode)
00057 {
00058     type_::UINT64 lret = teloc3000impl_:: NO_BOARD_ENABLED;
00059     constexpr type_::UINT64 MASK_DIGITAL = 0x18U;
00060     constexpr type_::UINT64 MASK_DAIOC   = 0x08U;
00061     constexpr type_::UINT64 MASK_REBOB   = 0x10U;
00062     type_::UINT64 lmain = maincode&MASK_DIGITAL;
00063     if(lmain == MASK_DIGITAL)
00064     {
00065         lret = (teloc3000impl_::DAIOD_ENABLED |
00066                 teloc3000impl_::REBO_ENABLED);
00067     }
00068     else
00069     {
00070         if(lmain == MASK_DAIOC)
00071             lret = (teloc3000impl_::DAIOD_ENABLED);
00072         else if(lmain == MASK_REBOB)
00073             lret = (teloc3000impl_::REBO_ENABLED);
```

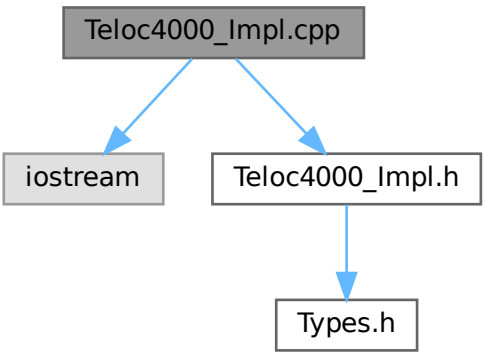
```
00074     }
00075     return((teloc3000impl_::eT3Code)lret);
00076 }
00077
00078 teloc3000impl_::eT3Code teloc3000impl_::setBUS(type_::UINT64 maincode)
00079 {
00080     type_::UINT64 lret = teloc3000impl_:: NO_BOARD_ENABLED;
00081     constexpr type_::UINT64 MASK_MVB = 0x40U;
00082     constexpr type_::UINT64 MASK_CAN = 0x80U;
00083     type_::UINT64 lmain = maincode&((MASK_MVB | MASK_CAN));
00084     //std::cout«"main_code = "«maincode«std::endl;
00085     //std::cout«"lmain = "«lmain«std::endl;
00086     if(lmain == (MASK_MVB | MASK_CAN))
00087     {
00088         lret = (teloc3000impl_:: MVB_ENABLED | teloc3000impl_:: CAN_ENABLED);
00089     }
00090     else
00091     {
00092         if(lmain == MASK_MVB)
00093             lret = (teloc3000impl_:: MVB_ENABLED);
00094         else if(lmain == MASK_CAN)
00095             lret = (teloc3000impl_:: CAN_ENABLED);
00096     }
00097     return((teloc3000impl_::eT3Code)lret);
00098 }
00099
00100 teloc3000impl_::eT3Code teloc3000impl_::setCPM(type_::UINT64 maincode)
00101 {
00102     type_::UINT64 lret = teloc3000impl_:: NO_BOARD_ENABLED;
00103     constexpr type_::UINT64 MASK_CPM = 0x200U;
00104     type_::UINT64 lmain = (maincode & MASK_CPM);
00105     return( (lmain == MASK_CPM) ? (teloc3000impl_::CPM_ENABLED) : (teloc3000impl_::NO_BOARD_ENABLED));
00106 }
00107
```

6.81 Teloc4000_Impl.cpp File Reference

file used to manage the methods for Teloc4000

```
#include "iostream"
#include "Teloc4000_Impl.h"
```

Include dependency graph for Teloc4000_Impl.cpp:



6.81.1 Detailed Description

file used to manage the methods for Teloc4000

Author

: Salvatore Muoio

Definition in file Teloc4000_Impl.cpp.

6.82 Teloc4000_Impl.cpp

Go to the documentation of this file.

```
00001
00006 #include "iostream"
00007 #include "Teloc4000_Impl.h"
00008
00009 using namespace teloc4000impl_;
00010
00011 teloc4000impl_::eT4Code teloc4000impl_::setSABO(type_::UINT64 maincode)
00012 {
00013     constexpr type_::UINT64 MASK_SABO = 0x20U;
00014     type_::UINT64 lmain = maincode&MASK_SABO;
00015     return((lmain == MASK_SABO) ? (teloc4000impl_:: SABOC_ENABLED) :
      (teloc4000impl_::NO_BOARD_ENABLED));
00016 }
00017
00018 teloc4000impl_::eT4Code teloc4000impl_::setTECA(type_::UINT64 maincode)
00019 {
00020     type_::UINT64 lret = teloc4000impl_:: NO_BOARD_ENABLED;
00021     constexpr type_::UINT64 MASK_IOCO = 0x04U;
00022     type_::UINT64 lmain = maincode&MASK_IOCO;
00023     if(lmain == MASK_IOCO)
00024     {
00025         if (setSABO(maincode) == teloc4000impl_:: SABOC_ENABLED)
00026             lret = teloc4000impl_::ONLY_DRSCA_ENABLED;
00027         else
00028             lret = teloc4000impl_::TECA_DRSCA_ENABLED;
00029     }
00030     else
00031         lret = teloc4000impl_::NO_BOARD_ENABLED;
00032     return((teloc4000impl_::eT4Code)lret);
00033 }
00034
00035 teloc4000impl_::eT4Code teloc4000impl_::setDIGITAL(type_::UINT64 maincode)
00036 {
00037     type_::UINT64 lret = teloc4000impl_:: NO_BOARD_ENABLED;
00038     constexpr type_::UINT64 MASK_DIGITAL = 0x18U;
00039     constexpr type_::UINT64 MASK_DAIOC   = 0x08U;
00040     constexpr type_::UINT64 MASK_REBOB   = 0x10U;
00041     type_::UINT64 lmain = maincode&MASK_DIGITAL;
00042     if(lmain == MASK_DIGITAL)
00043     {
00044         lret = (teloc4000impl_::DAIOD_ENABLED |
00045                 teloc4000impl_::DOCAA_ENABLED);
00046     }
00047     else
00048     {
00049         if(lmain == MASK_DAIOC)
00050             lret = (teloc4000impl_::DAIOD_ENABLED);
00051         else if(lmain == MASK_REBOB)
00052             lret = (teloc4000impl_::DOCAA_ENABLED);
00053     }
00054     return((teloc4000impl_::eT4Code)lret);
00055 }
00056
00057 teloc4000impl_::eT4Code teloc4000impl_::setBUS(type_::UINT64 maincode)
00058 {
00059     type_::UINT64 lret = teloc4000impl_:: NO_BOARD_ENABLED;
00060     constexpr type_::UINT64 MASK_MVB = 0x40U;
00061     constexpr type_::UINT64 MASK_CAN = 0x80U;
00062     type_::UINT64 lmain = maincode&((MASK_MVB | MASK_CAN));
00063     std::cout«"main_code = "«maincode«std::endl;
00064     std::cout«"lmain = "«lmain«std::endl;
00065     if(lmain == (MASK_MVB | MASK_CAN))
00066     {
00067         lret = (teloc4000impl_:: MVB_ENABLED | teloc4000impl_:: CAN_ENABLED);
00068     }
00069     else
00070     {
00071         if(lmain == MASK_MVB)
00072             lret = (teloc4000impl_:: MVB_ENABLED);
00073         else if(lmain == MASK_CAN)
```

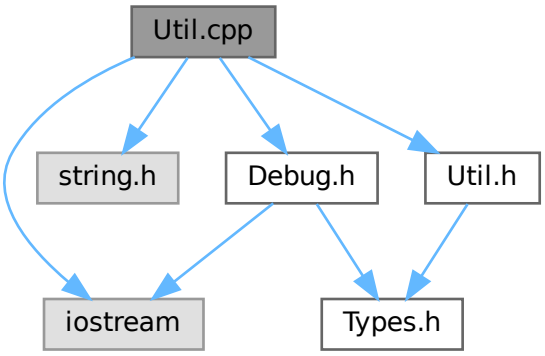
```
00074             lret = (teloc4000impl_:: CAN_ENABLED);
00075     }
00076     return((teloc4000impl_::eT4Code)lret);
00077 }
00078
00079 teloc4000impl_::eT4Code teloc4000impl_::setGPS(type_::UINT64 maincode)
00080 {
00081     type_::UINT64 lret = teloc4000impl_:: NO_BOARD_ENABLED;
00082     constexpr type_::UINT64 MASK_GPS = 0x100U;
00083     type_::UINT64 lmain = (maincode & MASK_GPS);
00084     return( (lmain == MASK_GPS) ? (teloc4000impl_::GPS_ENABLED) : (teloc4000impl_::NO_BOARD_ENABLED));
00085 }
00086
00087 teloc4000impl_::eT4Code teloc4000impl_::setCPM(type_::UINT64 maincode)
00088 {
00089     type_::UINT64 lret = teloc4000impl_:: NO_BOARD_ENABLED;
00090     constexpr type_::UINT64 MASK_CPM = 0x200U;
00091     type_::UINT64 lmain = (maincode & MASK_CPM);
00092     return( (lmain == MASK_CPM) ? (teloc4000impl_::CPM_ENABLED) : (teloc4000impl_::NO_BOARD_ENABLED));
00093 }
```

6.83 Util.cpp File Reference

In this file are implemented all methods util.

```
#include <iostream>
#include <string.h>
#include "Debug.h"
#include "Util.h"
```

Include dependency graph for Util.cpp:



6.83.1 Detailed Description

In this file are implemented all methods util.

Author

Salvatore Muoio

Definition in file Util.cpp.

6.84 Util.cpp

Go to the documentation of this file.

```
00001
00006 #include <iostream>
00007 #include <string.h>
00008 #include "Debug.h"
00009 #include "Util.h"
00010
00011 type_::ebool util_::charpointer_compare(const type_::CHAR *a, const type_::CHAR *b)
00012 {
00013     type_::ebool ret = type_::TRUE;
00014     for(int i = 0; strlen(a) < strlen(b) ? *a!= '\0' : *b!= '\0'; i++, *a++, *b++ )
00015     {
00016         if(*a != *b)
00017             ret = (type_::FALSE);
00018     }
00019     return(ret);
00020 }
00021
00022 type_::e_result util_::CheckArg(void *pArg)
00023 {
00024     return((pArg != 0) ? type_::RESULT_OK : type_::RESULT_POINTER_NOT_ADDRESSED);
00025 }
00026
00027 type_::UINT64 util_::ConverTelocCode2Num(std::string teloccode)
00028 {
00029 //  std::cout<<FUNCTION_NAME<<std::endl;
00030 //  std::cout<<teloccode<<endl;
00031     return( ((teloccode == "Teloc 1500") ? 0 : 1));
00032 }
```

Index