My Project

Generated by Doxygen 1.9.8

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 LSRecvUDPManager Class Reference

```
#include <LSRecvUDPManager.h>
```

Public Member Functions

- LSRecvUDPManager ()
- ∼LSRecvUDPManager ()
- bool LSSocketOpen (int port)
- bool receiveData (char ∗buffer, int bufferSize)
- void closeSocket ()

3.1.1 Constructor & Destructor Documentation

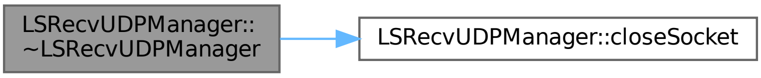
3.1.1.1 LSRecvUDPManager()

```
LSRecvUDPManager::LSRecvUDPManager ( )
```

3.1.1.2 ∼LSRecvUDPManager()

```
LSRecvUDPManager::∼LSRecvUDPManager ( )
```

Here is the call graph for this function:

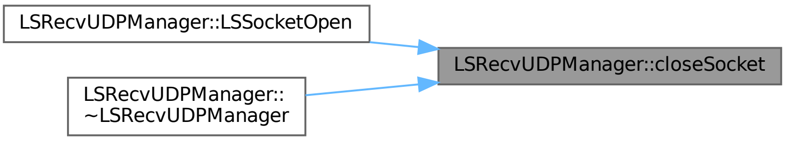


3.1.2 Member Function Documentation

3.1.2.1 closeSocket()

```
void LSRecvUDPManager::closeSocket ( )
```

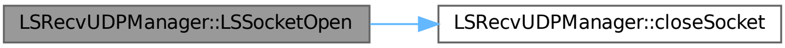
Here is the caller graph for this function:



3.1.2.2 LSSocketOpen()

```
bool LSRecvUDPManager::LSSocketOpen (
            int port )
```

Here is the call graph for this function:



3.1.2.3 receiveData()

```
bool LSRecvUDPManager::receiveData (
            char * buffer,
            int bufferSize )
```

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

- Simulator/UDPCommunicate/LSRecvUDPManager.h
- Simulator/UDPCommunicate/LSRecvUDPManager.cpp

3.2 MFRSendUDPManager Class Reference

```
#include <MFRSendUDPManager.h>
```

Public Member Functions

- MFRSendUDPManager ()
- ∼MFRSendUDPManager ()
- bool MFRSocketOpen (const std::string &ip, int port)
- bool sendData (const char ∗data, int dataSize)

3.2.1 Constructor & Destructor Documentation

3.2.1.1 MFRSendUDPManager()

```
MFRSendUDPManager::MFRSendUDPManager ( )
```

3.2.1.2 ∼MFRSendUDPManager()

```
MFRSendUDPManager::∼MFRSendUDPManager ( )
```

3.2.2 Member Function Documentation

3.2.2.1 MFRSocketOpen()

```
bool MFRSendUDPManager::MFRSocketOpen (
             const std::string & ip,
             int port )
```

3.2.2.2 sendData()

```
bool MFRSendUDPManager::sendData (
             const char ∗ data,
             int dataSize )
```

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

- Simulator/UDPCommunicate/MFRSendUDPManager.h
- Simulator/UDPCommunicate/MFRSendUDPManager.cpp

3.3 MissileInfo Struct Reference

```
#include <MissileInfo.h>
```

Public Attributes

- uint8_t mockType
- unsigned int id
- long long x
- long long y
- long long z
- double angle
- int speed

3.3.1 Member Data Documentation

3.3.1.1 angle

```
double MissileInfo::angle
```

3.3.1.2 id

```
unsigned int MissileInfo::id
```

3.3.1.3 mockType

```
uint8_t MissileInfo::mockType
```

3.3.1.4 speed

```
int MissileInfo::speed
```

3.3.1.5 x

```
long long MissileInfo::x
```

3.3.1.6 y

```
long long MissileInfo::y
```

3.3.1.7 z

```
long long MissileInfo::z
```

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

- Simulator/Mock/info/MissileInfo.h

3.4 MockMissile Class Reference

```
#include <MockMissile.h>
```

Public Member Functions

- MockMissile (const MissileInfo &missile_info, std::shared_ptr< MFRSendUDPManager > mfr_send_manager)
- ~MockMissile ()
- void updatePosMissile ()

3.4.1 Constructor & Destructor Documentation

3.4.1.1 MockMissile()

```
MockMissile::MockMissile (
            const MissileInfo & missile_info,
            std::shared_ptr< MFRSendUDPManager > mfr_send_manager )
```

3.4.1.2 ~MockMissile()

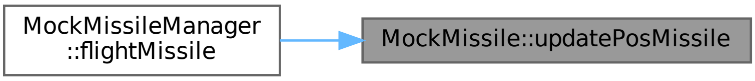
```
MockMissile::~MockMissile ( ) [inline]
```

3.4.2 Member Function Documentation

3.4.2.1 updatePosMissile()

```
void MockMissile::updatePosMissile ( )
```

Here is the caller graph for this function:


MockMissileManager ::flightMissile
MockMissile::updatePosMissile

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

- Simulator/Mock/MockMissile.h
- Simulator/Mock/MockMissile.cpp

3.5 MockMissileManager Class Reference

```
#include <MockMissileManager.h>
```

Public Member Functions

- MockMissileManager (std::shared_ptr< MockTargetManager > target_manager, std::shared_ptr< MFRSendUDPManager > mfr_send_manager)
- void flightMissile (const MissileInfo &MissileInfo)

3.5.1 Constructor & Destructor Documentation

3.5.1.1 MockMissileManager()

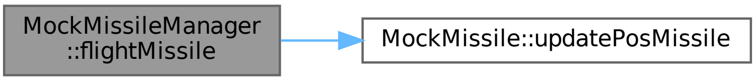
```
MockMissileManager::MockMissileManager (
            std::shared_ptr< MockTargetManager > target_manager,
            std::shared_ptr< MFRSendUDPManager > mfr_send_manager )
```

3.5.2 Member Function Documentation

3.5.2.1 flightMissile()

```
void MockMissileManager::flightMissile (
            const MissileInfo & MissileInfo )
```

Here is the call graph for this function:


MockMissileManager ::flightMissile
MockMissile::updatePosMissile

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

- Simulator/Mock/MockMissileManager.h
- Simulator/Mock/MockMissileManager.cpp

3.6 MockTarget Class Reference

```
#include <MockTarget.h>
```

Public Member Functions

- MockTarget (const TargetInfo &target_info, std::shared_ptr< MFRSendUDPManager > mfr_send_manager)
- ∼MockTarget ()
- void updatePos ()

3.6.1 Constructor & Destructor Documentation

3.6.1.1 MockTarget()

```
MockTarget::MockTarget (
         const TargetInfo & target_info,
         std::shared_ptr< MFRSendUDPManager > mfr_send_manager )
```

3.6.1.2 ∼MockTarget()

```
MockTarget::∼MockTarget ( )
```

3.6.2 Member Function Documentation

3.6.2.1 updatePos()

```
void MockTarget::updatePos ( )
```

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

- Simulator/Mock/MockTarget.h
- Simulator/Mock/MockTarget.cpp

3.7 MockTargetManager Class Reference

```
#include <MockTargetManager.h>
```

Public Member Functions

- MockTargetManager (std::shared_ptr< MFRSendUDPManager > mfr_send_manager)
- void RaedTargetIni ()
- void addTarget (std::shared_ptr< MockTarget > &target)
- void flitghtTarget ()

3.7.1 Constructor & Destructor Documentation

3.7.1.1 MockTargetManager()

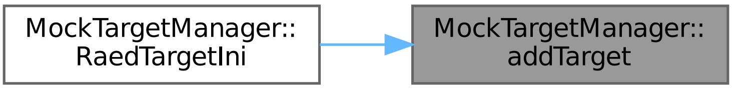
```
MockTargetManager::MockTargetManager (
         std::shared_ptr< MFRSendUDPManager > mfr_send_manager )
```

3.7.2 Member Function Documentation

3.7.2.1 addTarget()

```
void MockTargetManager::addTarget (
             std::shared_ptr< MockTarget > & target )
```

Here is the caller graph for this function:



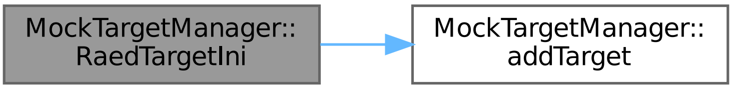
3.7.2.2 flitghtTarget()

```
void MockTargetManager::flitghtTarget ( )
```

3.7.2.3 RaedTargetIni()

```
void MockTargetManager::RaedTargetIni ( )
```

Here is the call graph for this function:



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

- Simulator/Mock/MockTargetManager.h
- Simulator/Mock/MockTargetManager.cpp

3.8 Simulator Class Reference

```
#include <Simulator.h>
```

Public Member Functions

- Simulator ()
- ∼Simulator ()
- bool init ()
- void start ()

3.8.1 Constructor & Destructor Documentation

3.8.1.1 Simulator()

```
Simulator::Simulator ( )
```

3.8.1.2 ∼Simulator()

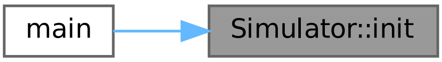
```
Simulator::∼Simulator ( )
```

3.8.2 Member Function Documentation

3.8.2.1 init()

```
bool Simulator::init ( )
```

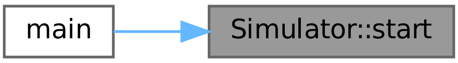
Here is the caller graph for this function:



3.8.2.2 start()

```
void Simulator::start ( )
```

Here is the caller graph for this function:



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

- Simulator/Simulator/Simulator.h
- Simulator/Simulator/Simulator.cpp

3.9 TargetInfo Struct Reference

```
#include <TargetInfo.h>
```

Public Attributes

- uint8_t mockType
- unsigned int id
- float x
- float y
- float z
- float angle
- int speed

3.9.1 Member Data Documentation

3.9.1.1 angle

```
float TargetInfo::angle
```

3.9.1.2 id

```
unsigned int TargetInfo::id
```

3.9.1.3 mockType

```
uint8_t TargetInfo::mockType
```

3.9.1.4 speed

```
int TargetInfo::speed
```

3.9.1.5 x

```
float TargetInfo::x
```

3.9.1.6 y

```
float TargetInfo::y
```

3.9.1.7 z

```
float TargetInfo::z
```

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

- Simulator/Mock/info/TargetInfo.h

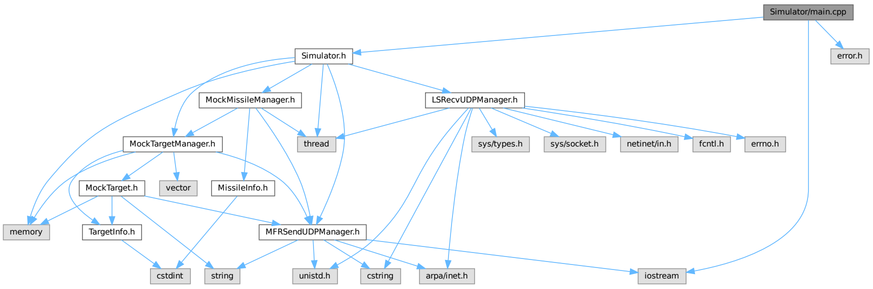
Chapter 4

File Documentation

4.1 Simulator/main.cpp File Reference

```
#include "Simulator.h"
#include <error.h>
#include <iostream>
```

Include dependency graph for main.cpp:



Functions

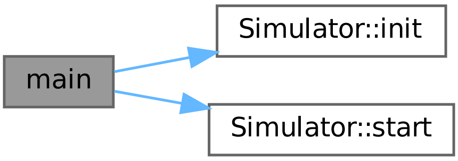
- int main ()

4.1.1 Function Documentation

4.1.1.1 main()

```
int main ( )
```

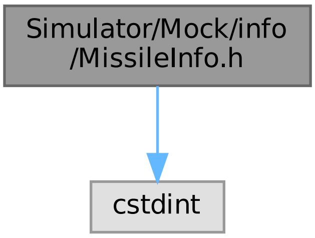
Here is the call graph for this function:



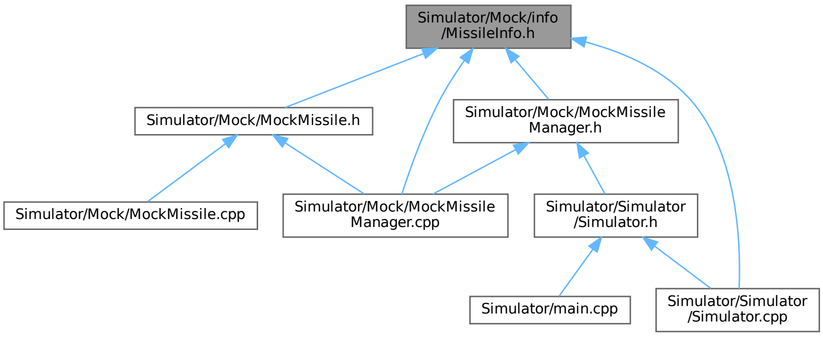
4.2 Simulator/Mock/info/MissileInfo.h File Reference

```
#include <cstdint>
```

Include dependency graph for MissileInfo.h:



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



Classes

- struct MissileInfo

4.3 MissileInfo.h

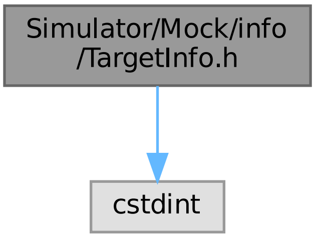
Go to the documentation of this file.

```
#pragma once
#include <cstdint>
#pragma pack(push, 1)
struct MissileInfo
{
    uint8_t mockType;
    unsigned int id;
    long long x;
    long long y;
    long long z;
    double angle;
    int speed;
};
#pragma pack(pop)
```

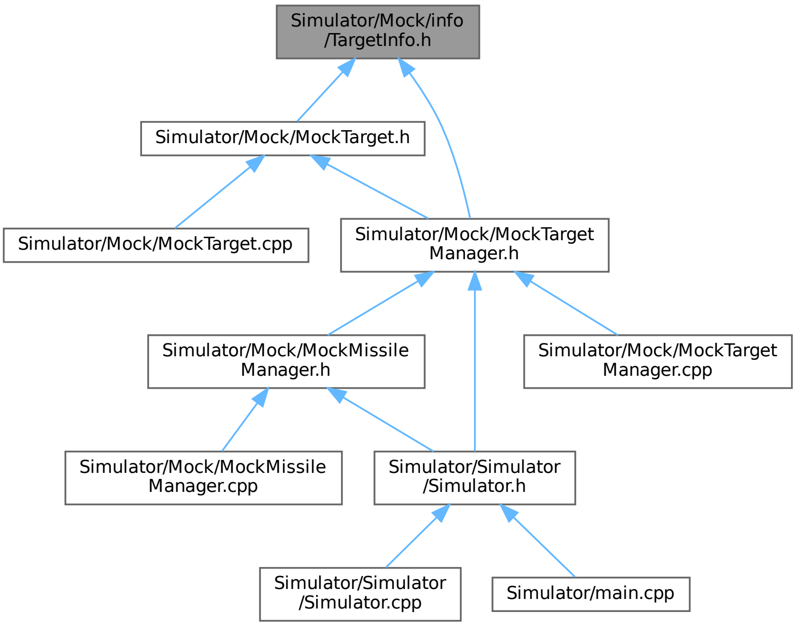
4.4 Simulator/Mock/info/TargetInfo.h File Reference

```
#include <cstdint>
```

Include dependency graph for TargetInfo.h:



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


Simulator/Mock/info
/TargetInfo.h
Simulator/Mock/MockTarget.h
Simulator/Mock/MockTarget.cpp
Simulator/Mock/MockTarget
Manager.h
Simulator/Mock/MockMissile
Manager.h
Simulator/Mock/MockTarget
Manager.cpp
Simulator/Mock/MockMissile
Manager.cpp
Simulator/Simulator
/Simulator.h
Simulator/Simulator
/Simulator.cpp
Simulator/main.cpp

Classes

- struct TargetInfo

4.5 TargetInfo.h

Go to the documentation of this file.

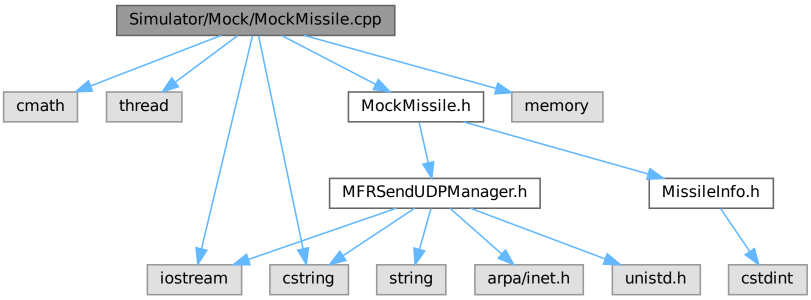
```
#pragma once
#include <cstdint>

#pragma pack(push, 1)
struct TargetInfo
{
    uint8_t mockType;
    unsigned int id;
    float x;
    float y;
    float z;
    float angle;
    int speed;
};
#pragma pack(pop)
```

4.6 Simulator/Mock/MockMissile.cpp File Reference

```
#include <cmath>
#include <thread>
#include <iostream>
#include <cstring>
```

```
#include <memory>
#include "MockMissile.h"
```

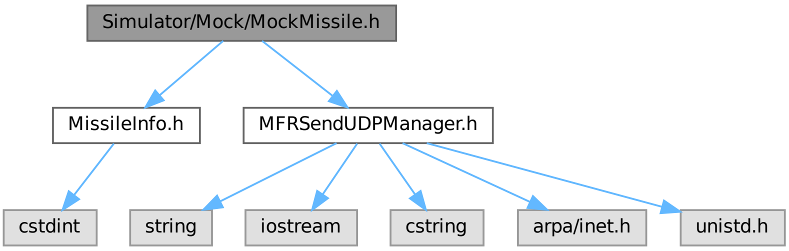
Include dependency graph for MockMissile.cpp:



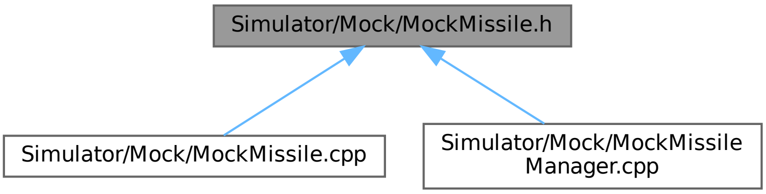
4.7 Simulator/Mock/MockMissile.h File Reference

```
#include "MissileInfo.h"
#include "MFRSendUDPManager.h"
```

Include dependency graph for MockMissile.h:



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


Simulator/Mock/MockMissile.h
Simulator/Mock/MockMissile.cpp
Simulator/Mock/MockMissile Manager.cpp

Classes

- class MockMissile

4.8 MockMissile.h

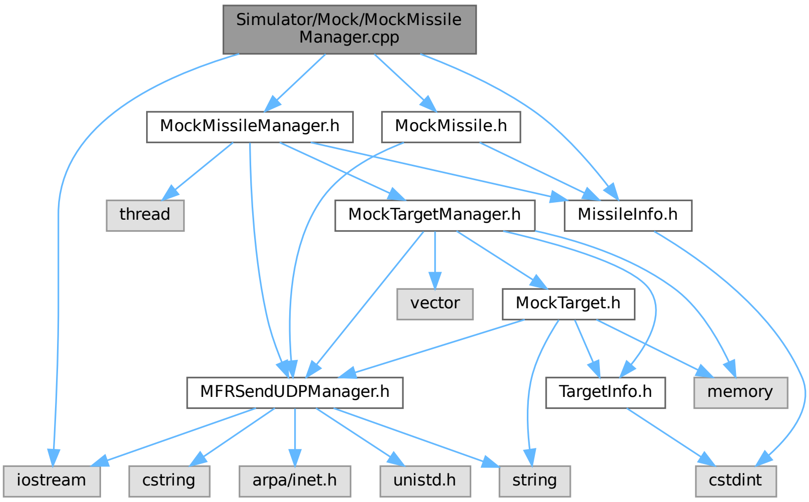
Go to the documentation of this file.

```
#pragma once
#include "MissileInfo.h"
#include "MFRSendUDPManager.h"

class MockMissile
{
public:
    MockMissile(const MissileInfo &missile_info, std::shared_ptr<MFRSendUDPManager> mfr_send_manager);
    ~MockMissile() {}

    void updatePosMissile();

private:
    MissileInfo missile_info_;
    std::shared_ptr<MFRSendUDPManager> mfr_send_manager_;

    void sendData();
};
```

4.9 Simulator/Mock/MockMissileManager.cpp File Reference

```
#include <iostream>
#include "MockMissileManager.h"
#include "MissileInfo.h"
#include "MockMissile.h"
```

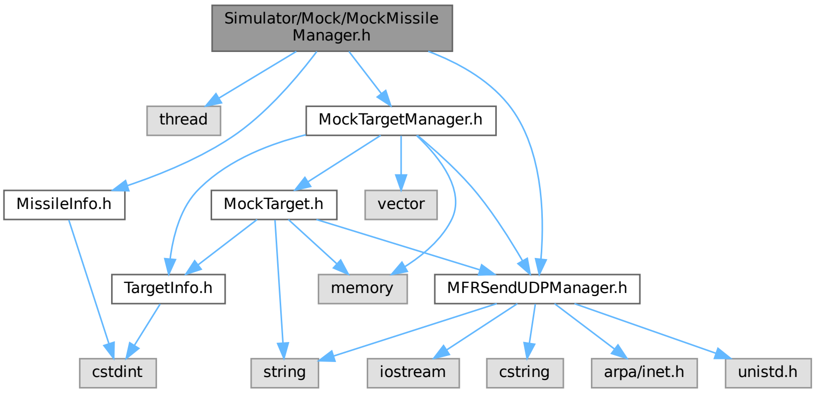
Include dependency graph for MockMissileManager.cpp:



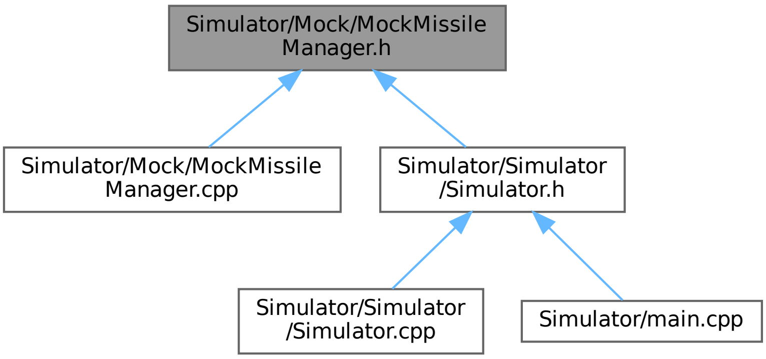
4.10 Simulator/Mock/MockMissileManager.h File Reference

```
#include <thread>
#include "MissileInfo.h"
#include "MockTargetManager.h"
#include "MFRSendUDPManager.h"
```

Include dependency graph for MockMissileManager.h:



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



Classes

- class MockMissileManager

4.11 MockMissileManager.h

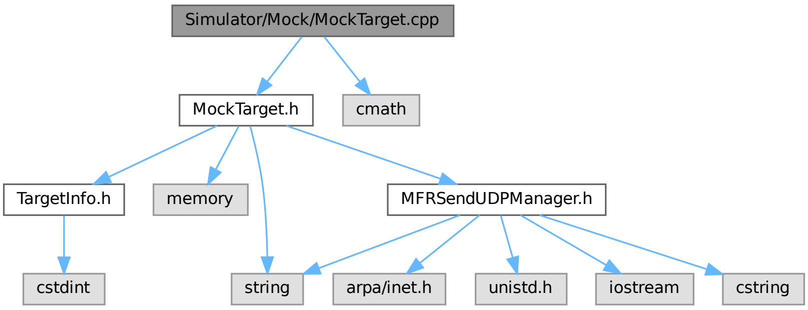
Go to the documentation of this file.

```
#ifndef MOCK_MISSILE_MANAGER_H
#define MOCK_MISSILE_MANAGER_H

#include <thread>

#include "MissileInfo.h"
#include "MockTargetManager.h"
#include "MFRSendUDPManager.h"

class MockMissileManager
{
private:
    std::shared_ptr<MockTargetManager> mock_target_manager_;
    std::shared_ptr<MFRSendUDPManager> mfr_send_manager_;

    bool is_flight_ = false;
    std::thread flight_thread_;

    int last_missile_id_ = 0;
    MissileInfo last_missile_info_;

    void updateMissileID();

public:
    MockMissileManager(std::shared_ptr<MockTargetManager> target_manager, std::shared_ptr<MFRSendUDPManager> mfr_send_manager);

    void flightMissile(const MissileInfo &MissileInfo);
};

#endif // MOCK_MISSILE_MANAGER_H
```

4.12 Simulator/Mock/MockTarget.cpp File Reference

```
#include "MockTarget.h"
#include <cmath>
```

Include dependency graph for MockTarget.cpp:

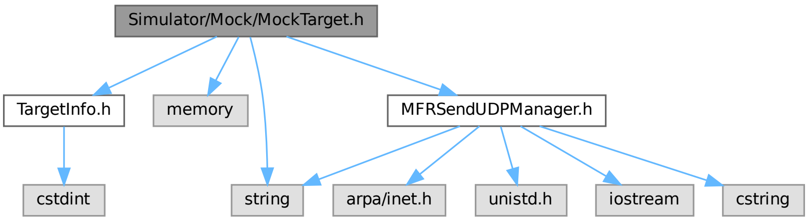


4.13 Simulator/Mock/MockTarget.h File Reference

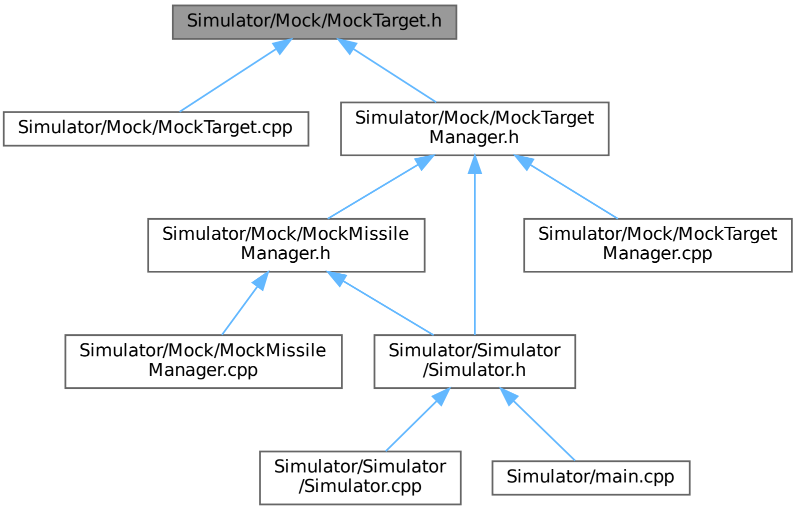
```
#include <string>
#include <memory>
#include "TargetInfo.h"
```

```
#include "MFRSendUDPManager.h"
```

Include dependency graph for MockTarget.h:



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


Simulator/Mock/MockTarget.h
Simulator/Mock/MockTarget.cpp
Simulator/Mock/MockTarget Manager.h
Simulator/Mock/MockMissile Manager.h
Simulator/Mock/MockTarget Manager.cpp
Simulator/Mock/MockMissile Manager.cpp
Simulator/Simulator /Simulator.h
Simulator/Simulator /Simulator.cpp
Simulator/main.cpp

Classes

- class MockTarget

4.14 MockTarget.h

Go to the documentation of this file.

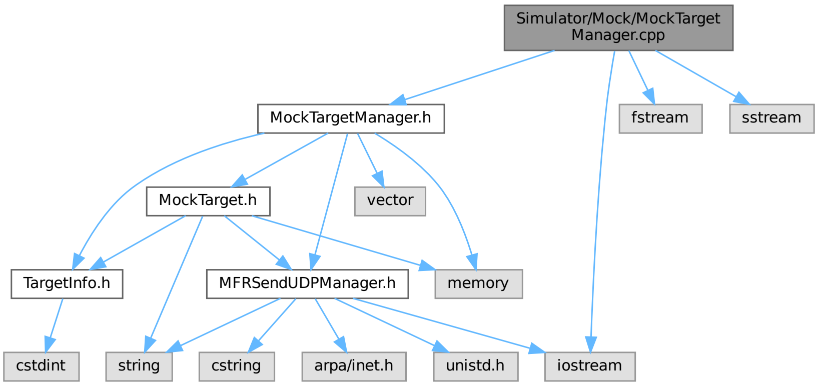
```
#ifndef MOCK_TARGET_H
#define MOCK_TARGET_H

#include <string>
#include <memory>

#include "TargetInfo.h"
#include "MFRSendUDPManager.h"

class MockTarget
```

```
{
public:
    MockTarget(const TargetInfo &target_info, std::shared_ptr<MFRSendUDPManager> mfr_send_manager);
    ~MockTarget();

    void updatePos();

private:
    TargetInfo target_info_;
    std::shared_ptr<MFRSendUDPManager> mfr_send_manager_ = nullptr;

    void sendData();
}
#endif // MOCK_TARGET_H
```

4.15 Simulator/Mock/MockTargetManager.cpp File Reference

```
#include "MockTargetManager.h"
#include <iostream>
#include <fstream>
#include <sstream>
```

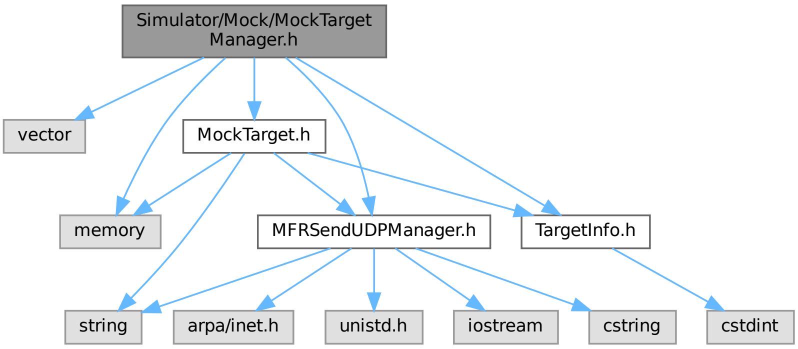
Include dependency graph for MockTargetManager.cpp:



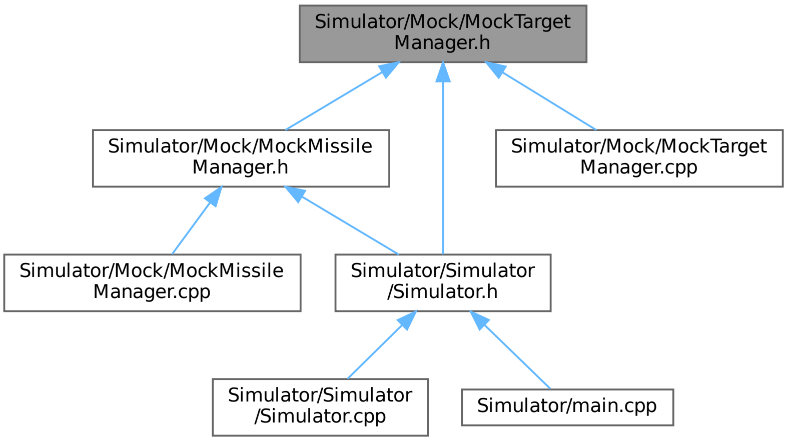
4.16 Simulator/Mock/MockTargetManager.h File Reference

```
#include <vector>
#include <memory>
#include "MockTarget.h"
#include "TargetInfo.h"
#include "MFRSendUDPManager.h"
```

Include dependency graph for MockTargetManager.h:



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


Simulator/Mock/MockTarget Manager.h
Simulator/Mock/MockMissile Manager.h
Simulator/Mock/MockTarget Manager.cpp
Simulator/Mock/MockMissile Manager.cpp
Simulator/Simulator /Simulator.h
Simulator/Simulator /Simulator.cpp
Simulator/main.cpp

Classes

- class MockTargetManager

4.17 MockTargetManager.h

Go to the documentation of this file.

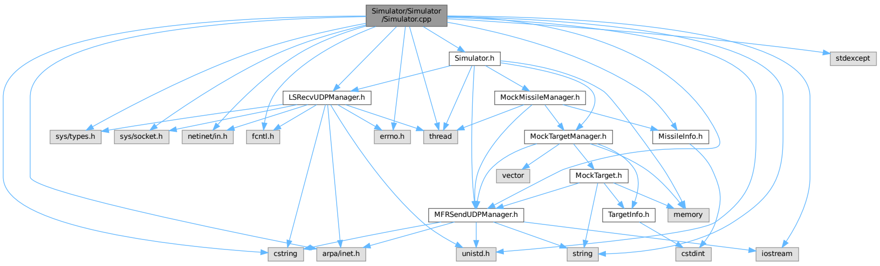
```
00001 #ifndef MOCK_TARGET_MANAGER_H
00002 #define MOCK_TARGET_MANAGER_H
00003
00004 #include <vector>
00005 #include <memory>
00006
00007 #include "MockTarget.h"
00008 #include "TargetInfo.h"
```

```

#include "MFRSendUDPManager.h"

class MockTargetManager
{
public:
    MockTargetManager(std::shared_ptr<MFRSendUDPManager> mfr_send_manager);
    void RaedTargetIni();
    void addTarget(std::shared_ptr<MockTarget> &target);
    void flitghtTarget();

private:
    std::vector<std::shared_ptr<MockTarget» targets;
    std::shared_ptr<MFRSendUDPManager> mfr_send_manager_;
};

#endif
```

4.18 Simulator/Simulator/Simulator.cpp File Reference

```
#include "Simulator.h"
#include <iostream>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <fcntl.h>
#include <errno.h>
#include <string>
#include <stdexcept>
#include <thread>
#include "MissileInfo.h"
#include "LSRecvUDPManager.h"
#include "MFRSendUDPManager.h"
```

Include dependency graph for Simulator.cpp:

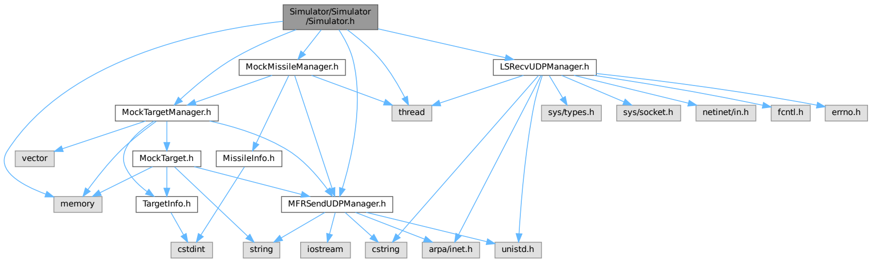


4.19 Simulator/Simulator/Simulator.h File Reference

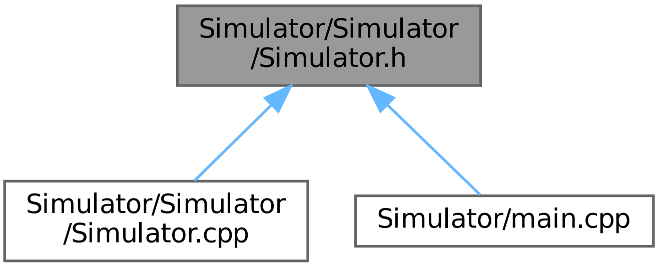
```
#include <thread>
#include <memory>
#include "LSRecvUDPManager.h"
#include "MFRSendUDPManager.h"
#include "MockTargetManager.h"
```

`#include "MockMissileManager.h"`

Include dependency graph for Simulator.h:



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



Classes

- class Simulator

4.20 Simulator.h

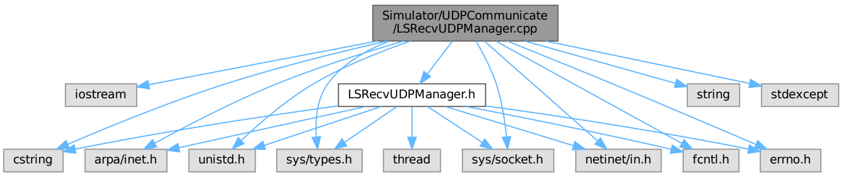
Go to the documentation of this file.

```
#include <thread>
#include <memory>

#include "LSRecvUDPManager.h"
#include "MFRSendUDPManager.h"
#include "MockTargetManager.h"
#include "MockMissileManager.h"

class Simulator
{
private:
    std::unique_ptr<LSRecvUDPManager> ls_recv_manager_;
    std::shared_ptr<MFRSendUDPManager> mfr_send_manager_;

    std::shared_ptr<MockTargetManager> mock_target_manager_;
    std::unique_ptr<MockMissileManager> mock_missile_manager_;

    std::thread recv_thread_;
    std::thread flight_target_thread_;

public:
    Simulator();
    ~Simulator();

    bool init();
    void start();
};
```

4.21 Simulator/UDPCommunicate/LSRecvUDPManager.cpp File Reference

```
#include <iostream>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <fcntl.h>
#include <errno.h>
#include <string>
#include <stdexcept>
#include "LSRecvUDPManager.h"
```

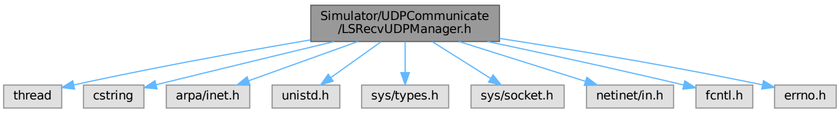
Include dependency graph for LSRecvUDPManager.cpp:



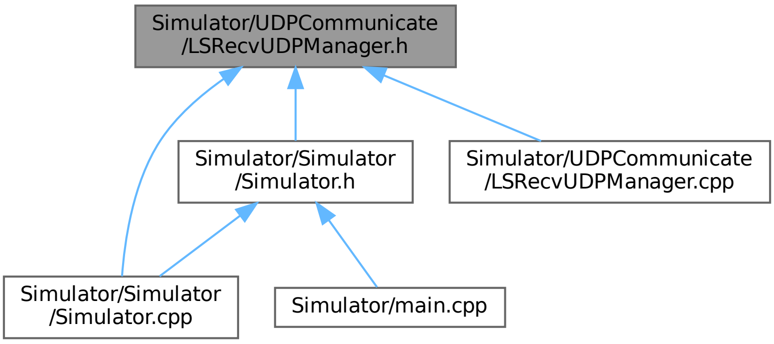
4.22 Simulator/UDPCommunicate/LSRecvUDPManager.h File Reference

```
#include <thread>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <fcntl.h>
#include <errno.h>
```

Include dependency graph for LSRecvUDPManager.h:



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



Classes

- class LSRecvUDPManager

4.23 LSRecvUDPManager.h

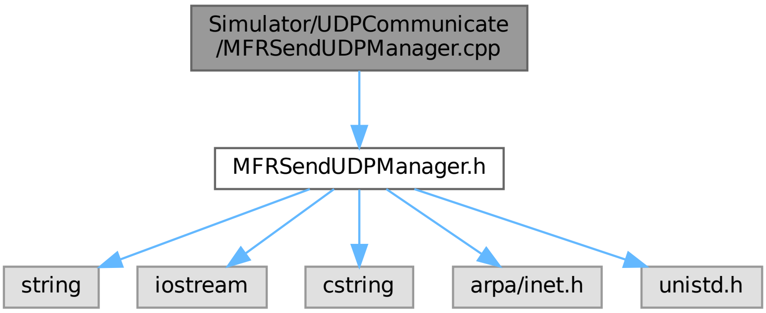
Go to the documentation of this file.

```
#ifndef LS_RECV_UDP_MANAGER_H
#define LS_RECV_UDP_MANAGER_H

#include <thread>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <fcntl.h>
#include <errno.h>

class LSRecvUDPManager
{
public:
    LSRecvUDPManager();
    ~LSRecvUDPManager();

    // Initialize the UDP socket
    bool LSSocketOpen(int port);

    // Receive data from the socket
    bool receiveData(char *buffer, int bufferSize);

    // Close the socket
    void closeSocket();

private:
    int ls_socket_;                   // Socket file descriptor
    struct sockaddr_in server_addr_; // Server address structure
    socklen_t addr_len_;             // Length of the address structure
    char buffer_[1024];              // Buffer for receiving data
    int buffer_size_;                // Size of the buffer
    int recv_len_;                   // Length of the received data
    std::thread recv_thread_;        // Thread for receiving data
};

#endif // LS_RECV_UDP_MANAGER_H
```

4.24 Simulator/UDPCommunicate/MFRSendUDPManager.cpp File Reference

```
#include "MFRSendUDPManager.h"
```

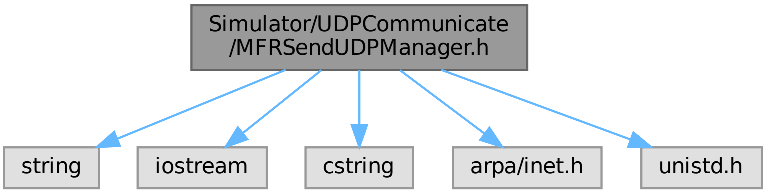
Include dependency graph for MFRSendUDPManager.cpp:



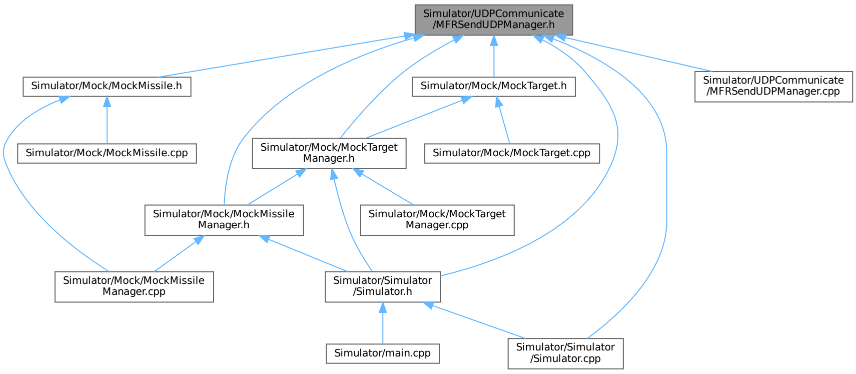
4.25 Simulator/UDPCommunicate/MFRSendUDPManager.h File Reference

```
#include <string>
#include <iostream>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>
```

Include dependency graph for MFRSendUDPManager.h:



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


Simulator/UDPCommunicate
/MFRSendUDPManager.h
Simulator/Mock/MockMissile.h
Simulator/Mock/MockTarget.h
Simulator/UDPCommunicate
/MFRSendUDPManager.cpp
Simulator/Mock/MockMissile.cpp
Simulator/Mock/MockTarget
Manager.h
Simulator/Mock/MockTarget.cpp
Simulator/Mock/MockMissile
Manager.h
Simulator/Mock/MockTarget
Manager.cpp
Simulator/Mock/MockMissile
Manager.cpp
Simulator/Simulator
/Simulator.h
Simulator/main.cpp
Simulator/Simulator
/Simulator.cpp

Classes

- class MFRSendUDPManager

4.26 MFRSendUDPManager.h

Go to the documentation of this file.

```
#ifndef MFR_SEND_UDP_MANAGER_H
#define MFR_SEND_UDP_MANAGER_H

#include <string>
#include <iostream>
#include <cstring>
#include <arpa/inet.h>
#include <unistd.h>

class MFRSendUDPManager
{
private:
    int mfr_socket_;
    struct sockaddr_in client_addr_;

public:
    MFRSendUDPManager(/* args */);
    ~MFRSendUDPManager();

    bool MFRSocketOpen(const std::string &ip, int port);
    bool sendData(const char *data, int dataSize);
};

#endif // MFR_SEND_UDP_MANAGER_H
```

Index