

# KBSW210707 Win32-

forbidden\_area,

- - 
  -
- 
- 
- 

- - Visual Studio 2010 SP1
  - Slamware Windows SDK:[Slamware Windows SDK](#)
  - RoboStudio():[Robostudio installer](#)
  - Sample Code:



Visual Studio

Visual Studio 2010SP1.Net FrameworkSP1

- - Slamware SDP mini
  - Slamware Slamware
  - Apollo/Ares/Athena

Win32-

## 1. samplesforbidden\_area StartUp project

📁 解决方案'samples' (15 个项目)

- > 📁 artifacts\_demo
- > 📁 composite\_map\_demo
- > 📁 configure\_network\_demo
- > 📁 **forbidden\_area**
- > 📁 get\_laser\_scan
- > 📁 get\_power\_status
- > 📁 get\_sensor\_value
- > 📁 go\_home\_to\_charge
- > 📁 move\_to\_spot
- > 📁 recover\_localization
- > 📁 robot\_health
- > 📁 rotation\_action\_demo
- > 📁 speed\_regulation

## 2. forbidden\_area, Slamware SDK includelib

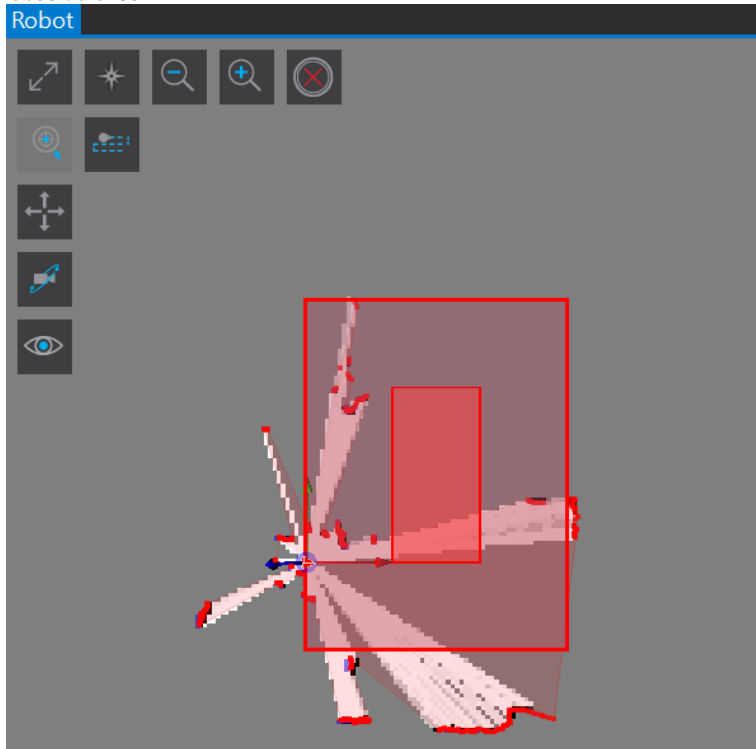
- > 通用属性
- ✓ 配置属性
  - 常规
  - 调试
  - VC++ 目录
  - > C/C++
  - > 链接器
  - > 清单工具
  - > XML 文档生成器
  - > 浏览信息
  - > 生成事件
  - > 自定义生成步骤
  - > 代码分析

- ### 3. forbidden\_area, Debuggingcommand Arguments 192.168.11.1

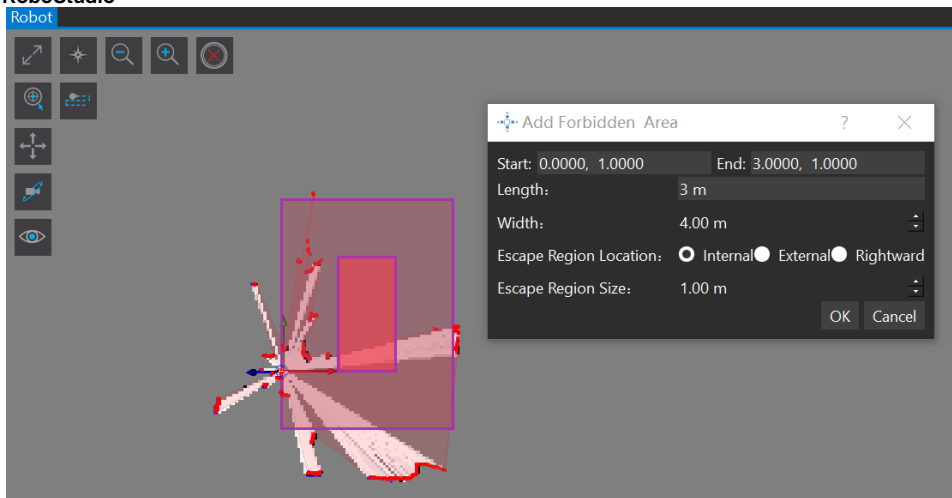
- 通用属性
- 配置属性
  - 常规
  - 调试
  - VC++ 目录
  - C/C++
  - 链接器
  - 清单工具
  - XML 文档生成器
  - 浏览信息
  - 生成事件
  - 自定义生成步骤
  - 代码分析

- #### 4. F5

## 5. robstudio10s



## 6. RoboStudio""



- 10s

```
SlamwareCorePlatform sdp = SlamwareCorePlatform::connect(ip_address, 1445);
std::cout <<"SDK Version: " << sdp.getSDKVersion() << std::endl;
std::cout <<"SDP Version: " << sdp.getSDPVersion() << std::endl;

/*add a forbidden area.          start point(0,1),          end point
(3,1),          width=2*2,          escape region size=1 */
rpos::core::ORectangleF area(rpos::core::Vector2f(0,1),rpos::core::Vector2f(3,1),2);

rpos::core::Metadata metadata;
std::string escapedistance="escape_distance";
```

```
std::string valueofescapedistance="1";
metadata.set(escapedistance,valueofescapedistance);

rpos::features::artifact_provider::RectangleArea rectanglearea;
rectanglearea.id=1;
rectanglearea.metadata=metadata;
rectanglearea.area=area;
rectanglearea.usage=ArtifactUsageForbiddenArea;

std::cout <<"Adding a forbidden area..."<< std::endl;
sdp.addRectangleArea(ArtifactUsageForbiddenArea,rectanglearea);

std::cout <<"Clearing all the forbidden area after 10s ..."<< std::endl;
boost::this_thread::sleep_for(boost::chrono::milliseconds(10000)); //sleep 10000ms
sdp.clearRectangleAreas(ArtifactUsageForbiddenArea);
```