KBSW180121 Win32-

move_to_spot,

-
  -
  -
-
-
-

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

    > Visual Studio
    >
    > Visual Studio 2010SP1.Net FrameworkSP1

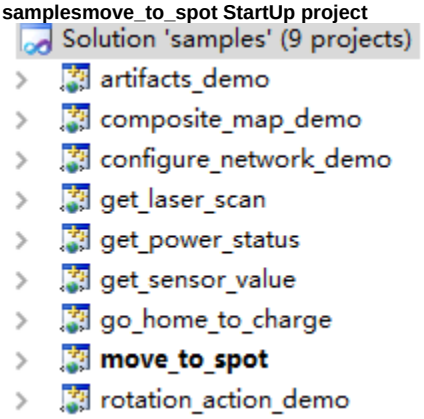
-

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

Win32-

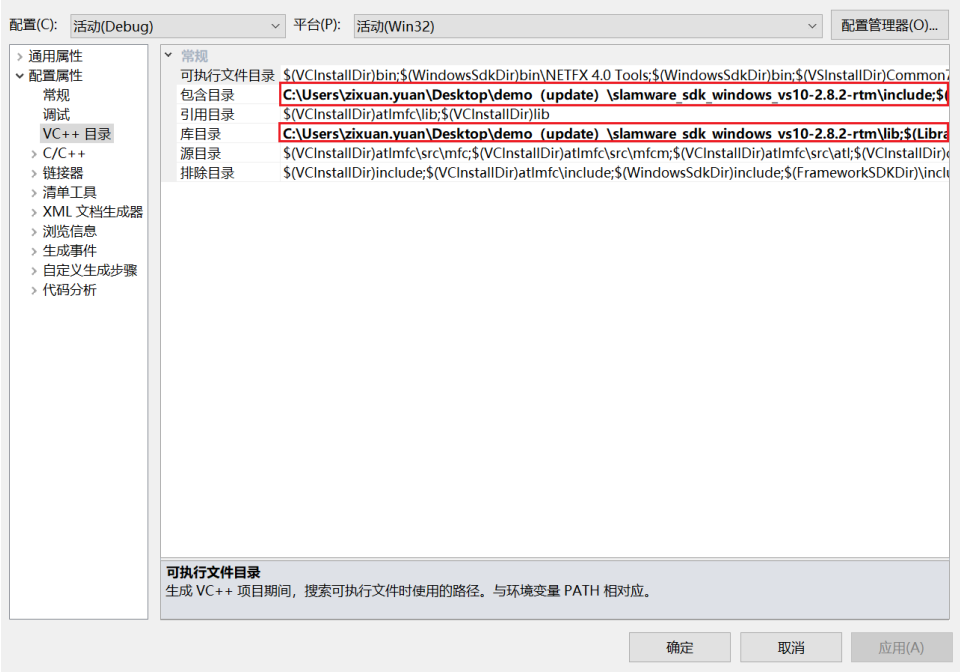
---

1. **samplesmove_to_spot StartUp project**

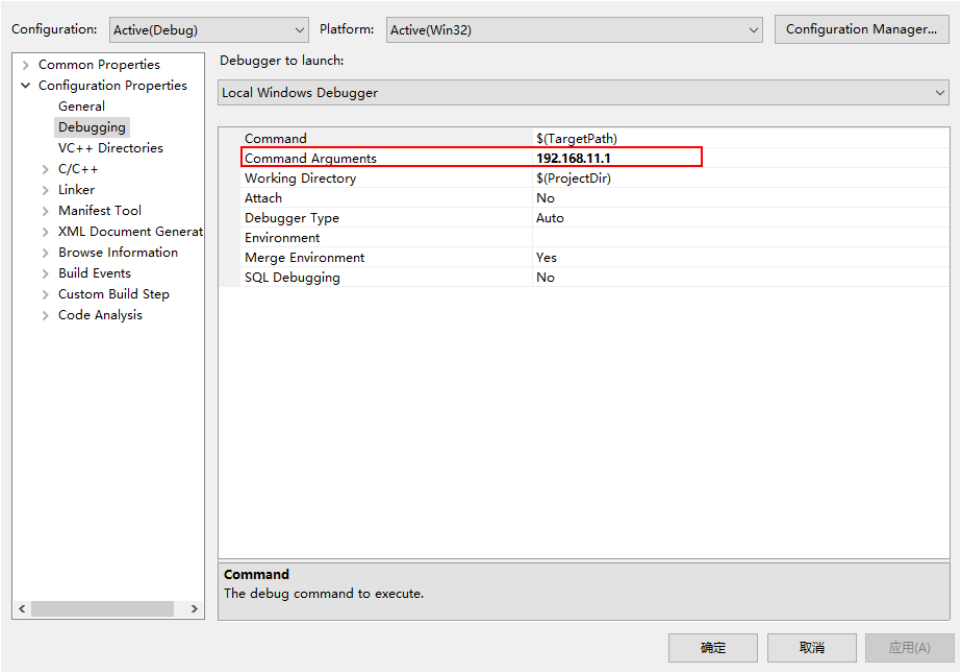


2. **move_to_spot, Slamware SDK includelib**

   > Slamware SDKincludelibVisual Studio



3. **move_to_spot, Debuggingcommand Arguments 192.168.11.1**
   move_to_spot <IP address>



4. **F5**
5. **Robostudio**
   Your browser does not support the HTML5 video element

- (2, 0) 20 0, 0

```
        SlamwareCorePlatform sdp = SlamwareCorePlatform::connect(argv[1], 1445);
                std::cout <<"SDK Version: " << sdp.getSDKVersion() << std::endl;
                std::cout <<"SDP Version: " << sdp.getSDPVersion() << std::endl;
                rpos::actions::MoveAction action = sdp.getCurrentAction();
                if (action)
                        action.cancel();
                //move to location (2, 0), not on virtual track
                rpos::features::motion_planner::MoveOptions options;
                options.flag = MoveOptionFlag(MoveOptionFlagMilestone | MoveOptionFlagPrecise);
                action = sdp.moveTo(rpos::core::Location(2, 0), options);
                action.waitUntilDone();
                if (action.getStatus() == rpos::core::ActionStatusError)
                        std::cout << "Action Failed: " << action.getReason() << std::endl;
                //draw a virtual track from (0, 0) to (2, 0), then move to (0, 0) via virtual track
                rpos::core::Line line(rpos::core::Point(0,0),rpos::core::Point(2,0));
                sdp.addLine(ArtifactUsageVirtualTrack, line);
                options.flag = MoveOptionFlag(MoveOptionFlagKeyPoints | MoveOptionFlagPrecise);
                action = sdp.moveTo(rpos::core::Location(0, 0), options);
                action.waitUntilDone();
                if (action.getStatus() == rpos::core::ActionStatusError)
                        std::cout << "Action Failed: " << action.getReason() << std::endl;
```