

FreeRTOS Windows Quick Start

Norman McEntire
norman.mcentire@gmail.com

Step 1. Go to FreeRTOS.org site

The screenshot shows the homepage of the FreeRTOS.org website. At the top, there is a header bar with a back arrow, forward arrow, and refresh icon. The URL https://freertos.org/index.html is displayed in the address bar. Below the address bar, there is a banner from AWS stating: "The FreeRTOS kernel is now an AWS open source project, and these pages are being updated accordingly. AWS is the MIT licensed Amazon FreeRTOS operating system, built on the FreeRTOS kernel". The main content area features the FreeRTOS logo (a green rounded rectangle with the word "freeRTOS" in white) and the text "Quality RTOS & Embedded Software". Below this, there are links for "About", "Contact", "Support", "FAQ", and "Download". A horizontal navigation bar below the main content includes links for "Quick Start", "Supported MCUs", "PDF Books", "Trace Tools", "Ecosystem", and "FreeRTOS International". On the left side, there is a sidebar with links for "Home", "MIT License", "FreeRTOS Books and Manuals", "FreeRTOS", and "FreeRTOS International". The right side features a large title "The FreeRTOS™ Kernel" and the subtitle "Market Leading, De-facto Standard and Cross Platform RTOS kernel."

Step 2. Click on Quick Start

The screenshot shows a web browser displaying the FreeRTOS website at <https://freertos.org/FreeRTOS-quick-start-guide.html>. The page header includes a message about the kernel becoming an MIT licensed AWS open source project. Below the header is the FreeRTOS logo. A red box highlights the "Quick Start" link in the top navigation bar, which is part of a blue menu bar also containing "Supported MCUs", "PDF Books", "Trace Tools", and "Ecosystem". To the left of the main content area is a sidebar with links to "Home", "MIT License", "FreeRTOS Books and Manuals", and a expanded section for "FreeRTOS" with "About FreeRTOS" and "Features / Getting Started...". The main content area features a large title "FreeRTOS Quick Start Guide" and a descriptive paragraph about the page's purpose.

The FreeRTOS kernel is now an **MIT** licensed **AWS open source project**, and

Quality RTOS & Embedded Software
[About](#) [Contact](#) [Support](#) [FAQ](#) [Download](#)

[Quick Start](#) | [Supported MCUs](#) | [PDF Books](#) | [Trace Tools](#) | [Ecosystem](#) |

[Home](#)
[MIT License](#)
[FreeRTOS Books and Manuals](#)
FreeRTOS
About FreeRTOS
[Features / Getting Started...](#)
[Quick Start Guide](#)

FreeRTOS Quick Start Guide

This page starts by describing how to get the RTOS running on your target. The "Next steps - further reading" section provides a set of links to enable you to answer common questions, and become an expert FreeRTOS user.

Step 3. Click on FreeRTOS Windows Port

| Quick Start | Supported MCUs | PDF Books | Trace Tools | Ecosystem |

[Home](#)

[MIT License](#)

[FreeRTOS Books and Manuals](#)

FreeRTOS

+ [About FreeRTOS](#)

- [Features / Getting Started...](#)

[Quick Start Guide](#)

[Tasks & Co-routines](#)

+ [More About Tasks...](#)

+ [Queues, Mutexes, Semaphores...](#)

+ [Direct To Task Notifications](#)

+ [Stream & Message Buffers](#)

+ [Software Timers](#)

[Event Groups \(or 'flags'\)](#)

+ [More About Co-routines](#)

FreeRTOS Quick Start Guide

This page starts by describing how to get the RTOS running on your target as quickly as possible. The "Next steps - further reading" section provides a set of links to enable you to further explore the FreeRTOS API, answer common questions, and become an expert FreeRTOS user.

Also see the [Getting Started With Simple FreeRTOS Projects](#) documentation, and the [FreeRTOS books](#). There is even a [FreeRTOS Windows port](#) to allow experimentation on a Windows host, using free tools, and without any special hardware requirements. It is encouraged to make use of the [configASSERT\(\)](#) macro.

Step 4. Click on Getting Started with Simple FreeRTOS Projects

Quick Start | Supported MCUs | PDF Books | Trace Tools | Ecosystem | Email List+ 

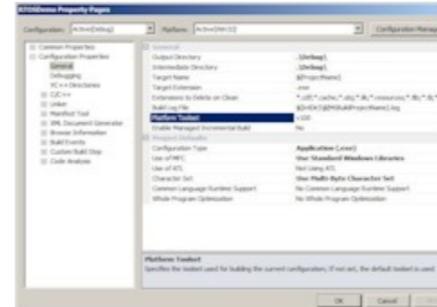
[Home](#)
[MIT License](#)
[FreeRTOS Books and Manuals](#)
FreeRTOS
+ [About FreeRTOS](#)
+ [Features / Getting Started...](#)
+ [More Advanced...](#)
[Creating a New Project](#)
[FreeRTOSConfig.h](#)
[Trace Features](#)
[Low Power Support](#)
[Run Time Stats](#)
[Blocking on Multiple Objects](#)
[Deferred Interrupt Handling](#)
[Static Vs Dynamic Memory](#)
[Memory Management](#)
[Memory Protection Support](#)
[Stack Overflow Protection](#)
[Hook Functions](#)
[Thread Local Storage Pointers](#)
+ [How FreeRTOS Works](#)
+ [RAM Constrained Design Tips](#)
[Porting Guide](#)
[Windows Simulator](#)
[Posix/Linux Simulator](#)
[Legacy Trace Facility](#)
+ [Demo Projects](#)

FreeRTOS Windows Port

For Visual Studio or Eclipse and MingW

[RTOS Ports]

A Note For Users of FreeRTOS V9.0.0! The Win32 project in the FreeRTOS V9.0.0 distribution uses Visual Studio 2015 Community Edition in place of Visual Studio 2010 Express Edition. The project can still be opened in Visual Studio 2010, but the compiler version referenced from the project's options must be manually updated before the project can be built. The compiler version is set using the "Platform Toolset" option highlighted in the screen shot on the right. Click the screen shot to enlarge.



Preamble - for beginners

If you are new to FreeRTOS then it is recommended to start by viewing the [Getting Started With Simple FreeRTOS Projects](#) documentation (which also describes how to use the FreeRTOS Windows port), before viewing this page.

Step 5. Observe (1) Simple blinky demos

[Home](#)
[MIT License](#)
[FreeRTOS Books and Manuals](#)
FreeRTOS
+ [About FreeRTOS](#)
+ [Features / Getting Started...](#)
+ [More Advanced...](#)
Demo Projects
Standard Demos
Basic Demos
Hardware Independent
Modifying a Demo
TCP/IP Demos
+ [Supported Devices & Demos](#)
+ [API Reference](#)
+ [Contact & Support](#)
+ [FreeRTOS Interactive!](#)

[Quick Start Guide](#)
[Support Forum](#)
[Download Source](#)

Getting Started with Simple FreeRTOS Projects

[See also the [Quick Start Guide](#) and the [hardware independent](#) starter functions provided.]

Simply Blinky Projects

The most recent FreeRTOS demos can be configured to build either a simple blinky starter application, or a comprehensive test and demo application:

1. Simple blinky demos

Blinky demos are intended for beginners. They normally create just two [tasks](#), one [queue](#), and sometimes also a [software timer](#). Their functionality is contained in a single C source file called `main_blinky.c`.

2. Comprehensive demos

Comprehensive demos demonstrate and test a lot of FreeRTOS features, including [tasks](#), [direct to task notifications](#), [queues](#), semaphores, recursive semaphores, [software timers](#), and more. Comprehensive demos create a lot of tasks and use a lot of API functions, so are not recommended for beginners.

Step 6. Scroll down until you see “Try It Now, Using the Windows Port”

The sidebar contains the following links:

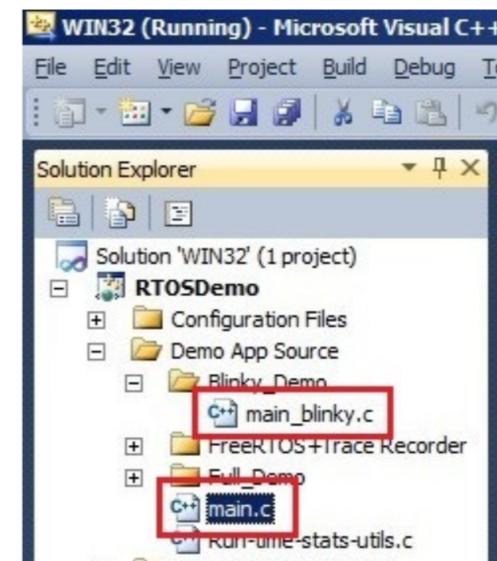
- Trace & Visualisation: Tracealyzer for FreeRTOS
- CLI: Command line interface
- WolfSSL SSL / TLS: Networking security protocols
- RTOS Training: Delivered online or on-site
- IO: read(), write(), ioctl() interface
- FreeRTOS+ Lab Projects**
- FreeRTOS+POSIX: POSIX threading API
- FreeRTOS+FAT: Thread aware file system

Try It Now, Using the Windows Port

No hardware yet? Don't worry - you can run a simple blinky demo in a Windows environment using free tools and the FreeRTOS Windows port, although the FreeRTOS Windows port will not exhibit true real time behaviour.

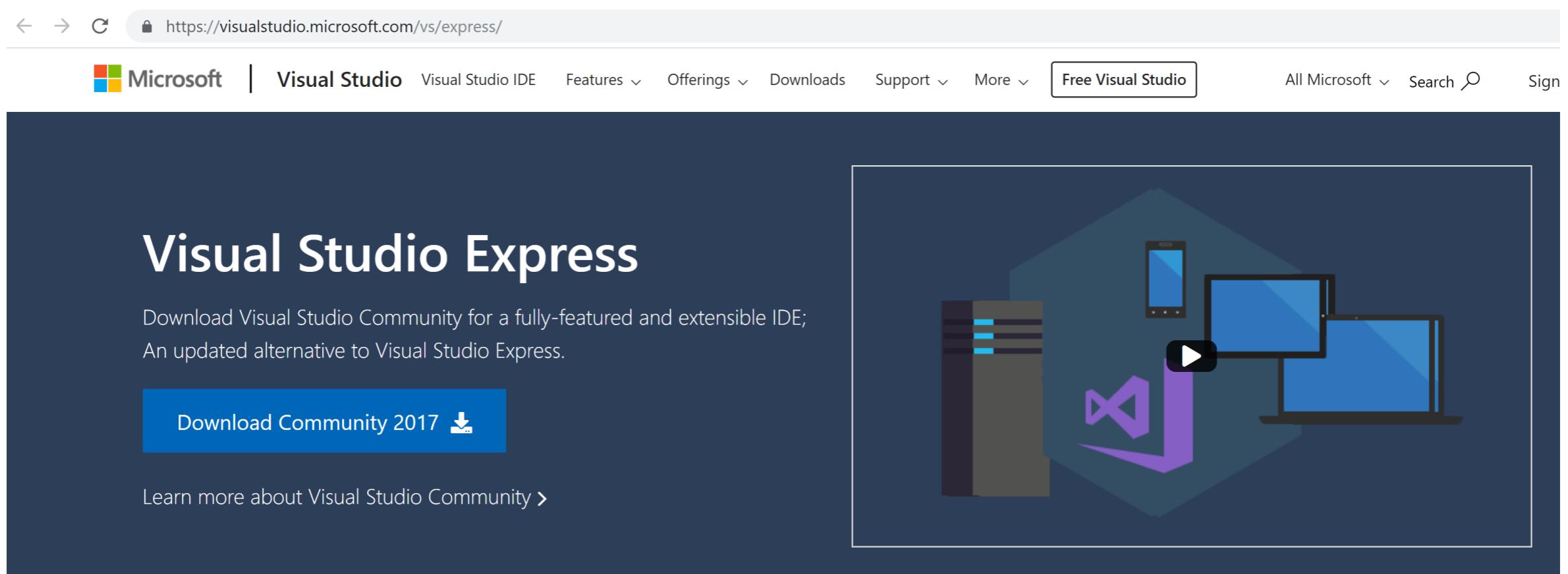
If you are a beginner, then don't read the [main documentation page for the Windows demo](#) yet, or pay any attention to the comprehensive demo that can be built by the same project (the comprehensive demo for the Windows port is **very** comprehensive as it is used as a development and test platform!) - just start by using the simple instructions below:

1. If you don't already have it installed, download and install the [free](#) version of Microsoft Visual Studio.
2. If you have not done so already, [download](#) and unzip the official FreeRTOS distribution.
3. Start Visual Studio, then use the File|Open|Project/Solution menu item to open the Win32.sln solution file, which is located in the FreeRTOS/Demo/WIN32-MSVC directory of the official FreeRTOS distribution.
4. Find the definition of mainCREATE_SIMPLE_BLINKY_DEMO_ONLY at the top of main.c, and make sure it is set to 1.



Step 7. If you have not already done so, download free Visual Studio Express

<https://visualstudio.microsoft.com/vs/express/>



The screenshot shows the Microsoft Visual Studio Express landing page. At the top, there is a navigation bar with links for Microsoft, Visual Studio, Visual Studio IDE, Features, Offerings, Downloads, Support, More, Free Visual Studio (which is highlighted), All Microsoft, Search, and Sign In. The main content area has a dark blue background. On the left, the text "Visual Studio Express" is displayed in large white font. Below it, a subtitle reads "Download Visual Studio Community for a fully-featured and extensible IDE; An updated alternative to Visual Studio Express." A blue button labeled "Download Community 2017" with a download icon is present. To the right, there is a large graphic featuring the Visual Studio logo (a purple 'X' icon) and various icons representing different development tools like a smartphone, a laptop, and a server.

Step 8. Download and Unzip the FreeRTOS code: <https://freertos.org/a00104.html>

The screenshot shows a web browser displaying the FreeRTOS website at <https://freertos.org/a00104.html>. The page header includes a navigation bar with back, forward, and search icons, and a URL bar showing the current address. A banner at the top states: "The FreeRTOS kernel is now an **MIT** licensed **AWS open source project**, and these pages are being updated acc...". The main content area features the FreeRTOS logo and navigation links for Quality RTOS & Embedded Software, including About, Contact, Support, FAQ, and Download. Below this is a horizontal menu with links for Quick Start, Supported MCUs, PDF Books, Trace Tools, Ecosystem, and social media links for Email List and Twitter. On the left, there is a sidebar with links for Home, MIT License, FreeRTOS Books and Manuals, and sections for FreeRTOS and FreeRTOS Interactive. The main content area is titled "RTOS Source Code Download Instructions" and provides steps for downloading the latest FreeRTOS versions, mentioning the announcements mailing list and SourceForge releases. A red box highlights the "Download Source Code and Projects" button.

The FreeRTOS kernel is now an **MIT** licensed **AWS open source project**, and these pages are being updated acc...

Quality RTOS & Embedded Software

[About](#) [Contact](#) [Support](#) [FAQ](#) [Download](#)

| [Quick Start](#) | [Supported MCUs](#) | [PDF Books](#) | [Trace Tools](#) | [Ecosystem](#) |

[Email List](#) [Twitter](#)

[Home](#)
[MIT License](#)
[FreeRTOS Books and Manuals](#)
+ [FreeRTOS](#)
+ [FreeRTOS Interactive!](#)

[Quick Start Guide](#)
[Support Forum](#)
[↓ Download Source ↓](#)

[FreeRTOS+ Ecosystem](#)
[FreeRTOS+TCP:](#)
Thread safe TCP/IP stack

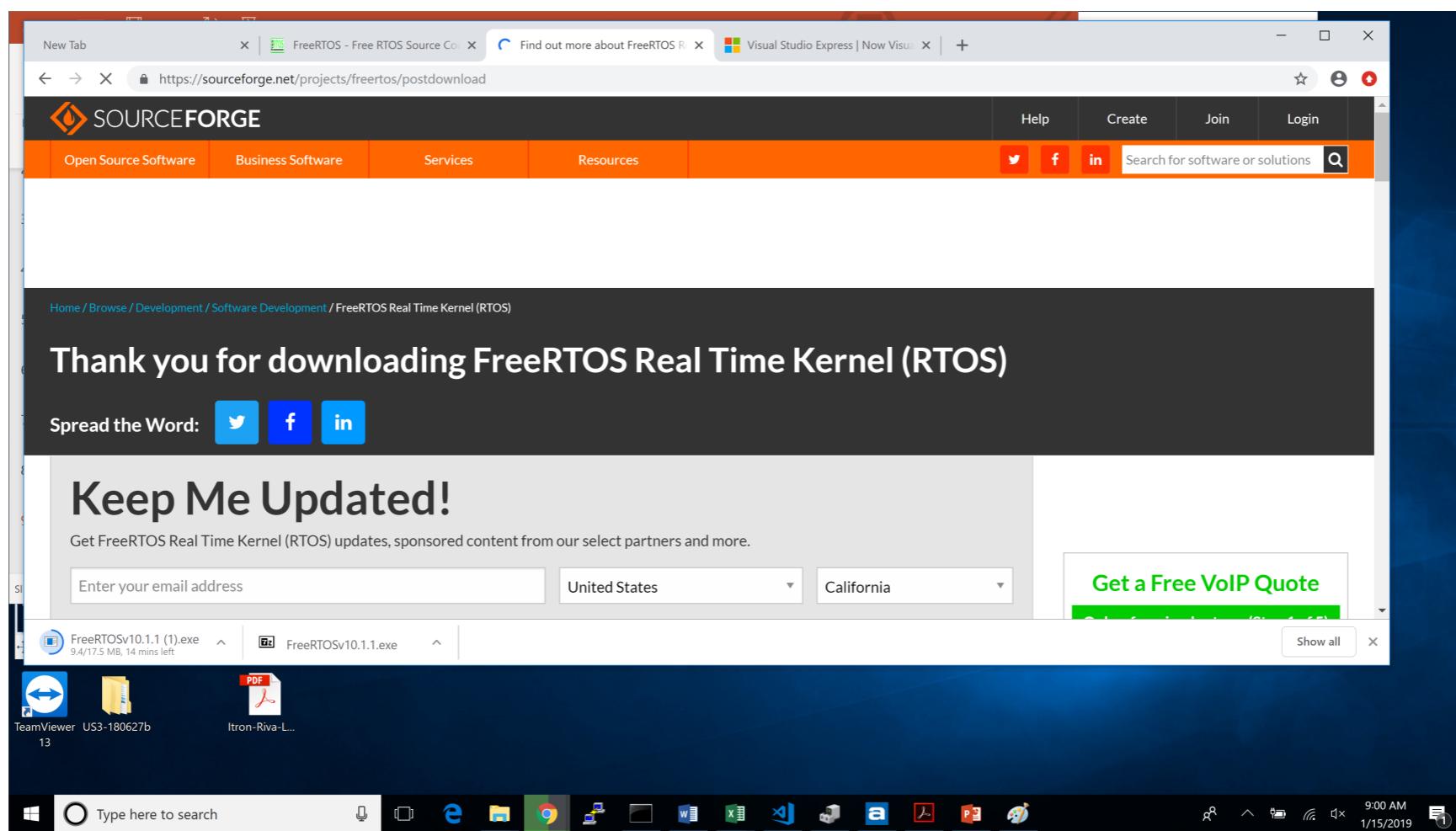
RTOS Source Code Download Instructions

Follow the steps below to download the latest FreeRTOS versions:

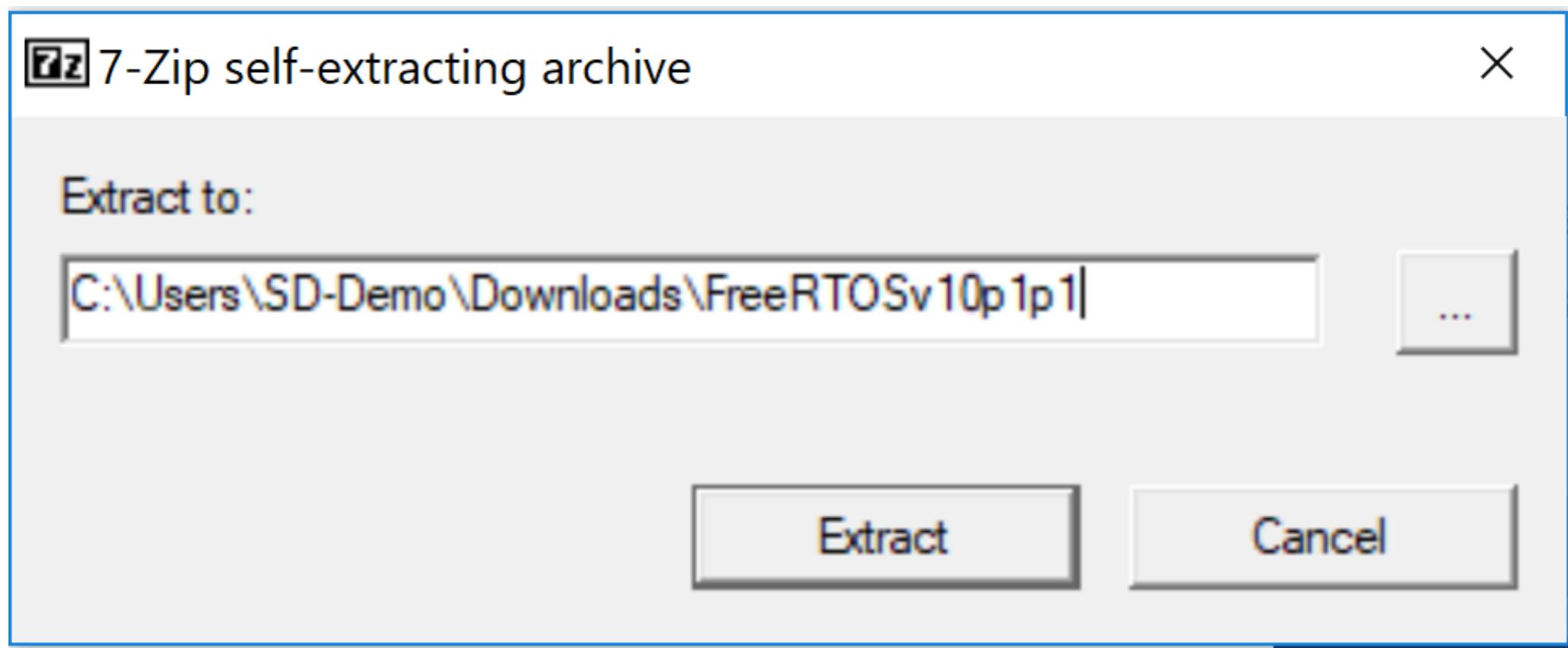
1. Please keep up to date by joining the [announcements mailing list](#) for infrequent comprehensive notifications. We respect your privacy, so do not provide email addresses to any third party. Every email sent contains unsubscribe instructions.
2. Download the [latest official release](#) or a [previous release](#) from SourceForge, both of which are available as a [standard zip file \(.zip\)](#), and as a [self extracting zip file \(.exe\)](#). Alternatively obtain the source files [directly from SVN](#).

[Download Source Code and Projects](#)

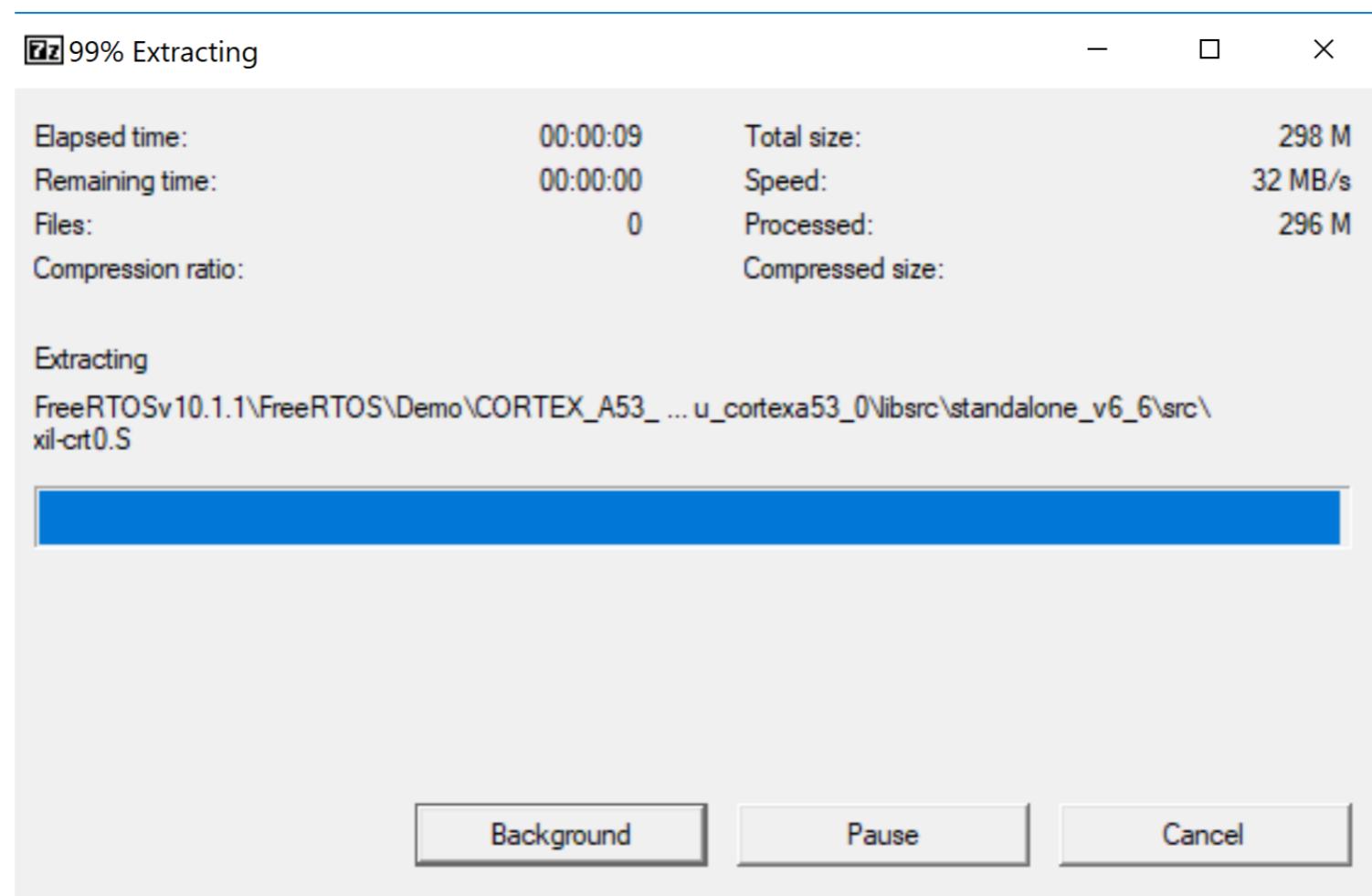
Step 9. Results of Download: FreeRTOSv10.1.1.exe – A Self-Extracting ZIP file



Step 10. Run FreeRTOSV10.1.1.exe

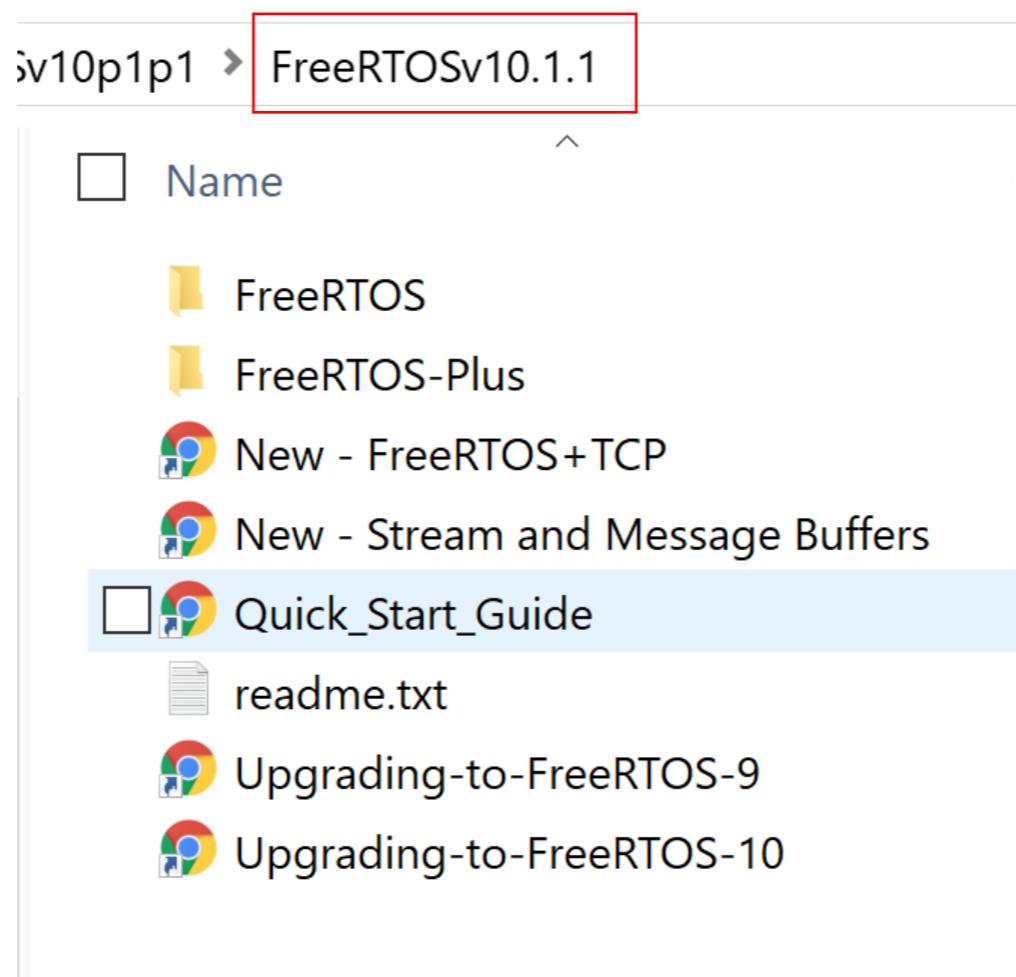


Step 11. Extraction in progress...



Step 12. Tour of Top-Level Directory

FreeRTOSv10.1.1 Directory



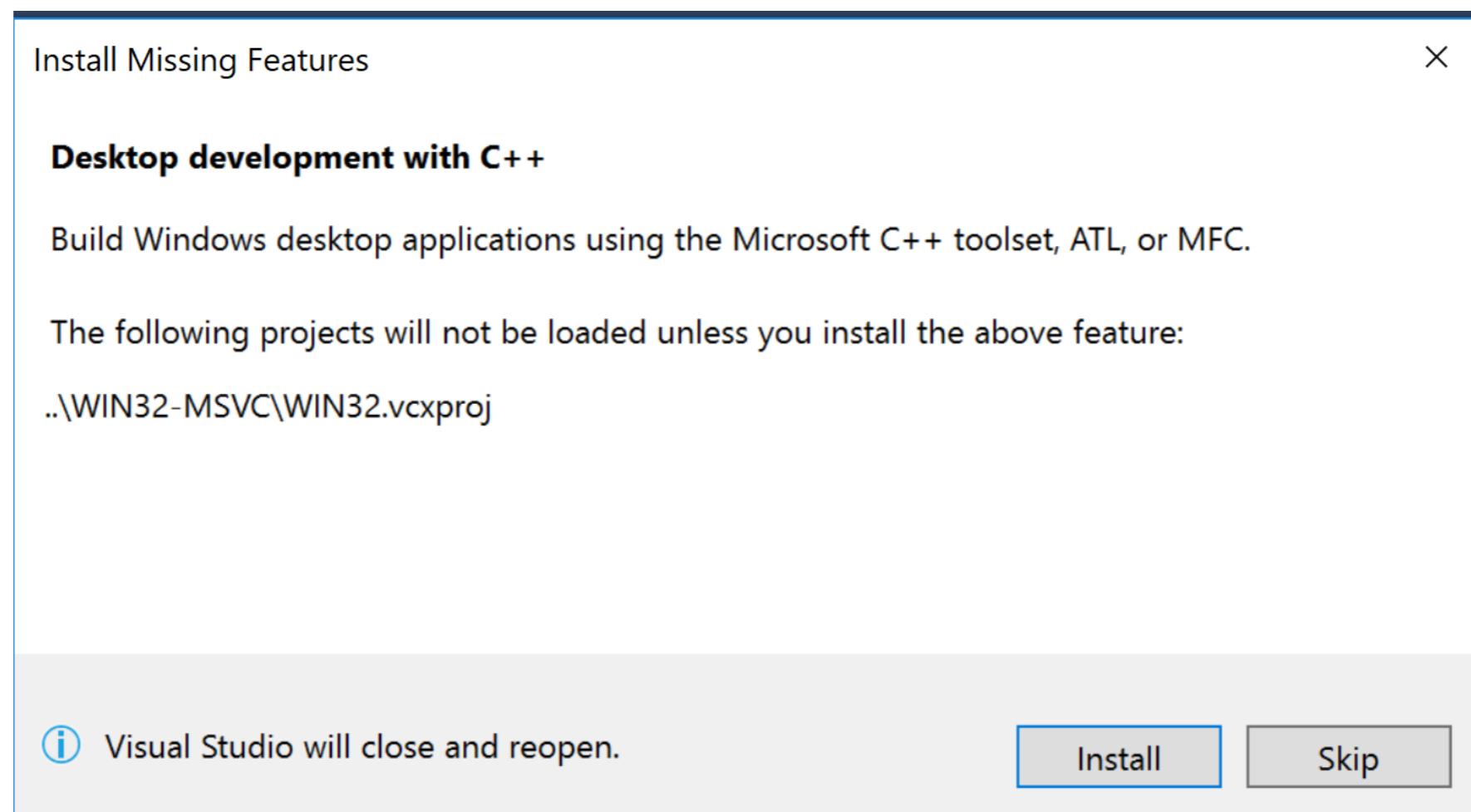
Step 13. Tour of FreeRTOS/Demo/WIN32-MSVC

Sv10p1p1 > FreeRTOSv10.1.1 > FreeRTOS > Demo > WIN32-MSVC				
<input type="checkbox"/> Name	Date modified	Type	Size	
.vs	9/7/2018 1:58 PM	File folder		
TraceRecorderConfiguration	9/7/2018 1:58 PM	File folder		
FreeRTOSConfig.h	9/7/2018 2:51 PM	H File	6 KB	
main.c	9/7/2018 2:51 PM	C File	17 KB	
main_blinky.c	9/7/2018 2:51 PM	C File	12 KB	
main_full.c	9/7/2018 2:51 PM	C File	28 KB	
Run-time-stats-utils.c	9/7/2018 2:51 PM	C File	4 KB	
WIN32.sln	5/7/2017 11:19 AM	Visual Studio Solu...	2 KB	
WIN32.vcxproj	11/27/2017 4:36 P...	VC++ Project	10 KB	
WIN32.vcxproj.filters	11/22/2017 5:49 P...	VC++ Project Filte...	10 KB	

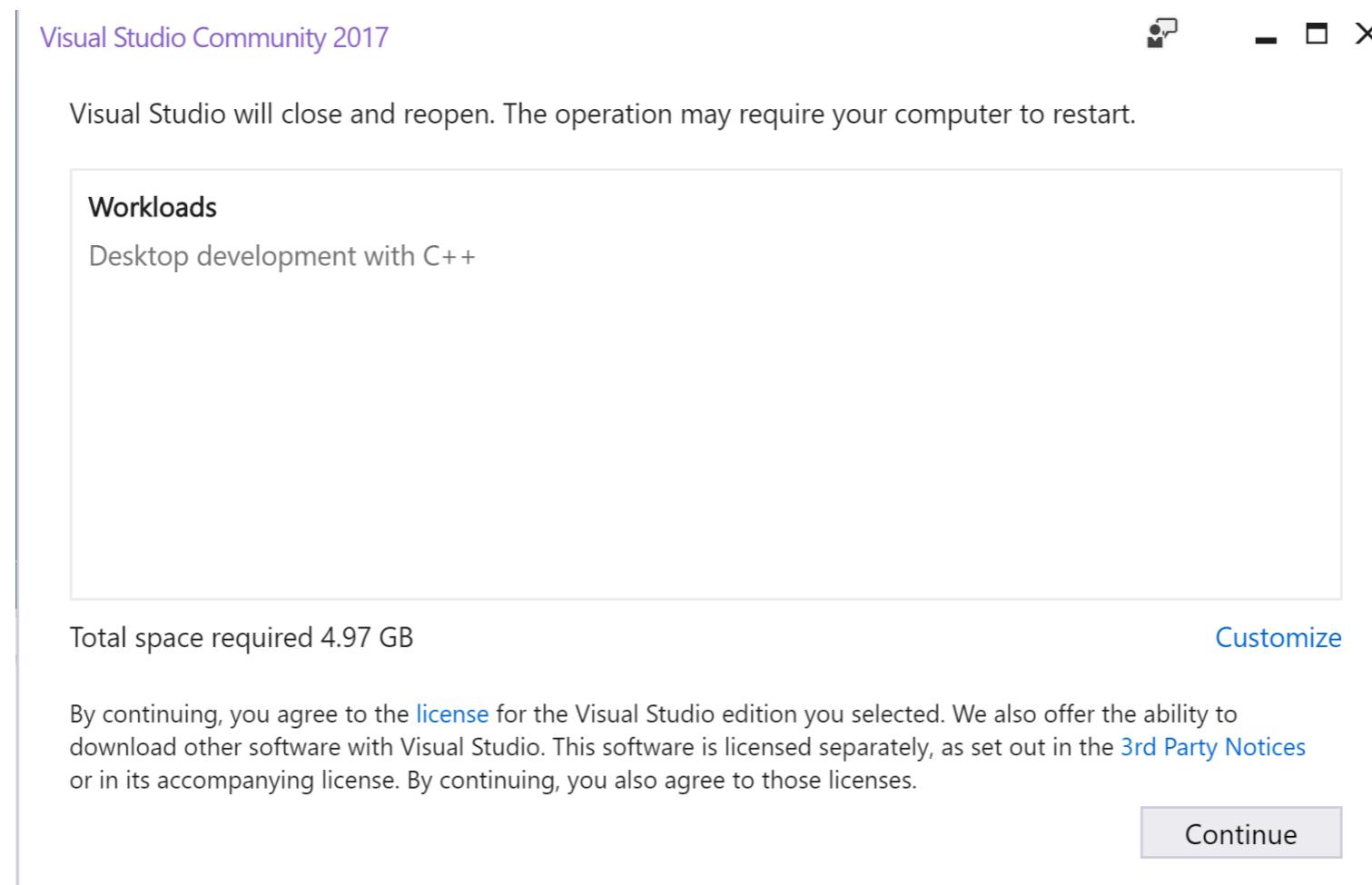
Step 14. Double-click on WIN32.sln into Visual Studion

Sv10p1p1 > FreeRTOSv10.1.1 > FreeRTOS > Demo > WIN32-MSVC				
<input type="checkbox"/> Name	Date modified	Type	Size	
.vs	9/7/2018 1:58 PM	File folder		
TraceRecorderConfiguration	9/7/2018 1:58 PM	File folder		
FreeRTOSConfig.h	9/7/2018 2:51 PM	H File	6 KB	
main.c	9/7/2018 2:51 PM	C File	17 KB	
main_blinky.c	9/7/2018 2:51 PM	C File	12 KB	
main_full.c	9/7/2018 2:51 PM	C File	28 KB	
Run-time-stats-utils.c	9/7/2018 2:51 PM	C File	4 KB	
WIN32.sln	5/7/2017 11:19 AM	Visual Studio Solu...	2 KB	
WIN32.vcxproj	11/27/2017 4:36 P...	VC++ Project	10 KB	
WIN32.vcxproj.filters	11/22/2017 5:49 P...	VC++ Project Filte...	10 KB	

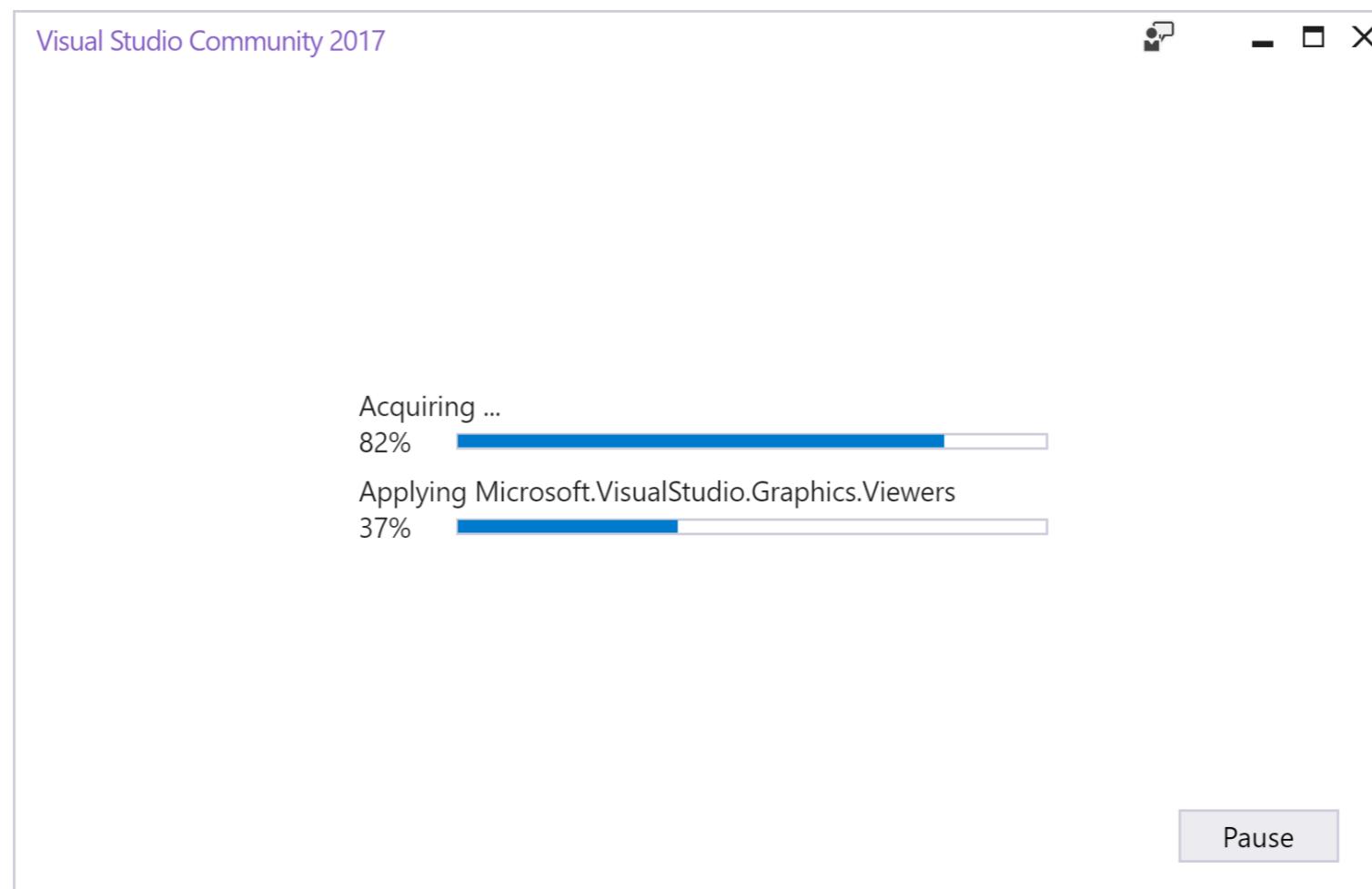
Step 15a. (If you do not have C++ installed into Visual Studio)



Step 15b. (Continuing the install)



Step 15c. (continuing the install)



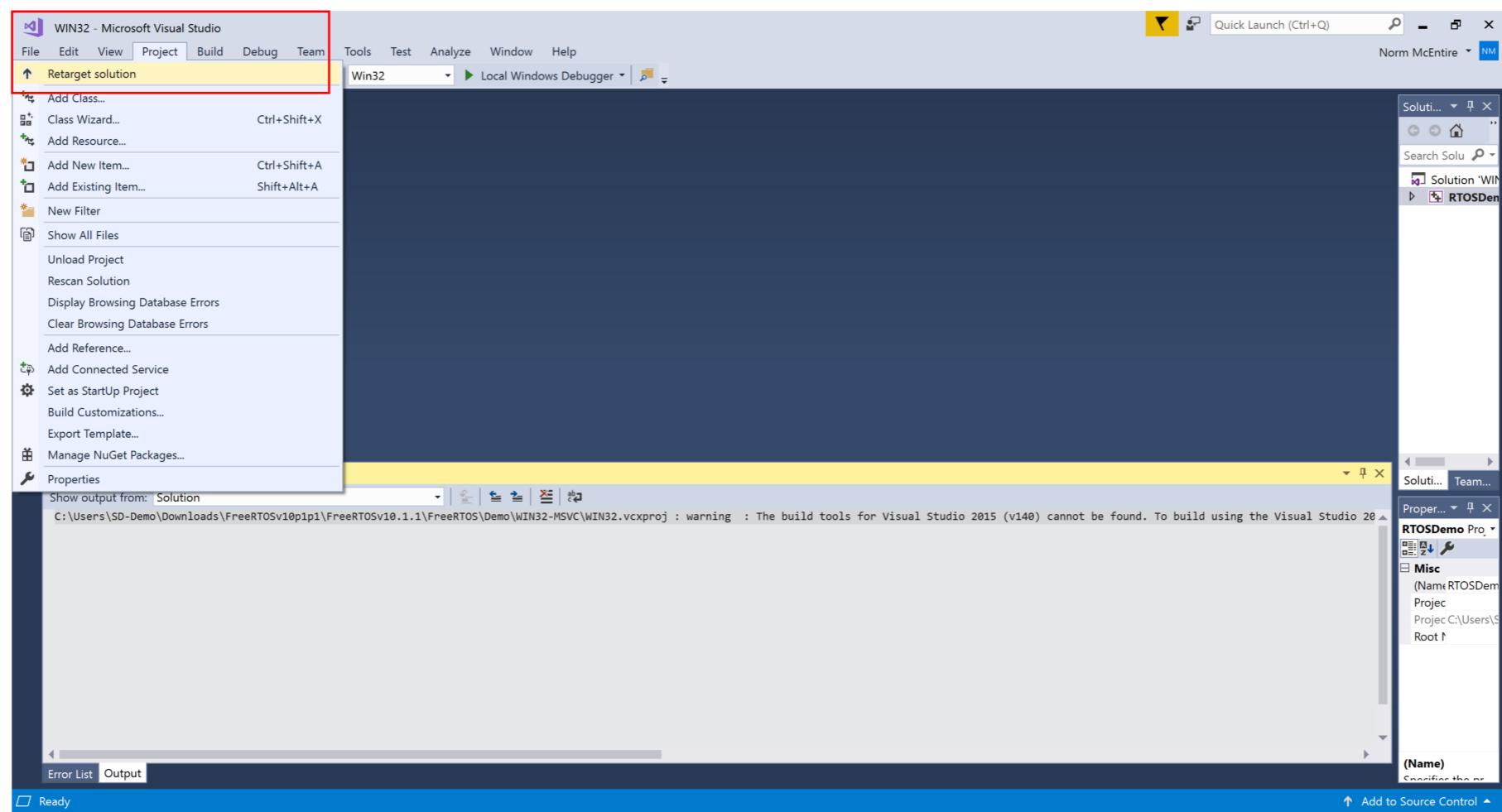
Step 15d. (Install Complete)



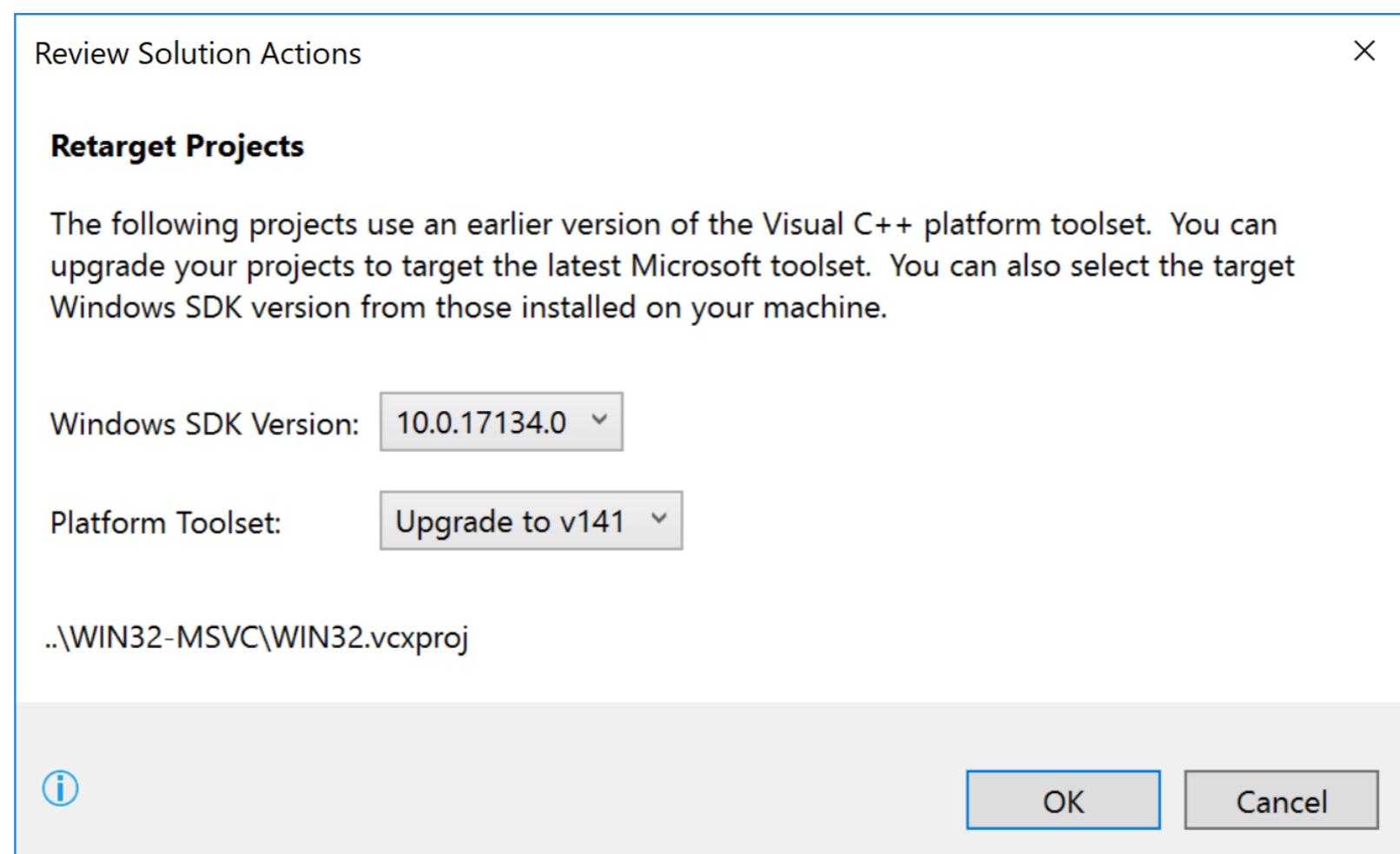
Step 16. Observe Project Opened in Visual Studio – Error Message

- C:\Users\SD-Demo\Downloads\FreeRTOSv10p1p1\FreeRTOSv10.1.1\FreeRTOS\Demo\WIN32-MSVC\WIN32.vcxproj : warning : The build tools for Visual Studio 2015 (v140) cannot be found. To build using the Visual Studio 2017 (v141) build tools, either click the Project menu or right-click the solution, and then select "Retarget Solution". Install Visual Studio 2015 (v140) to build using the Visual Studio 2015 (v140) build tools.

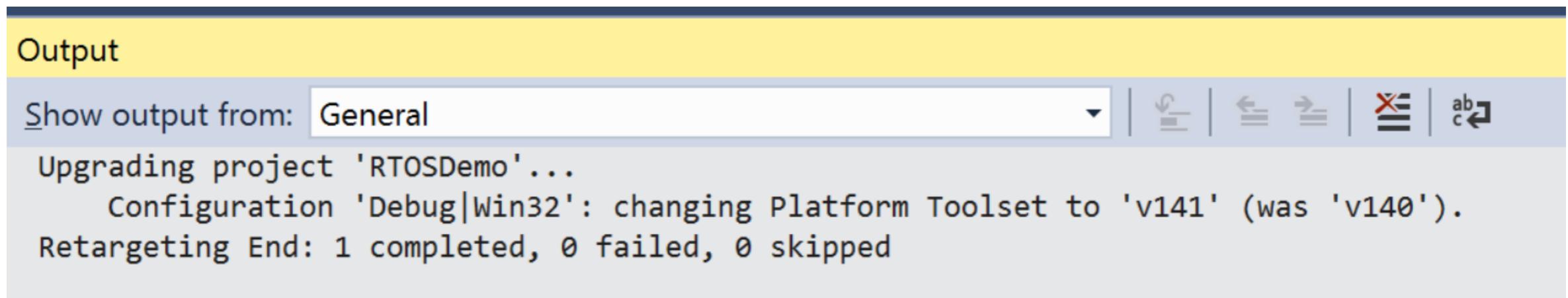
Step 17a. Retargeting Solution



Step 17b. Review Solution Actions



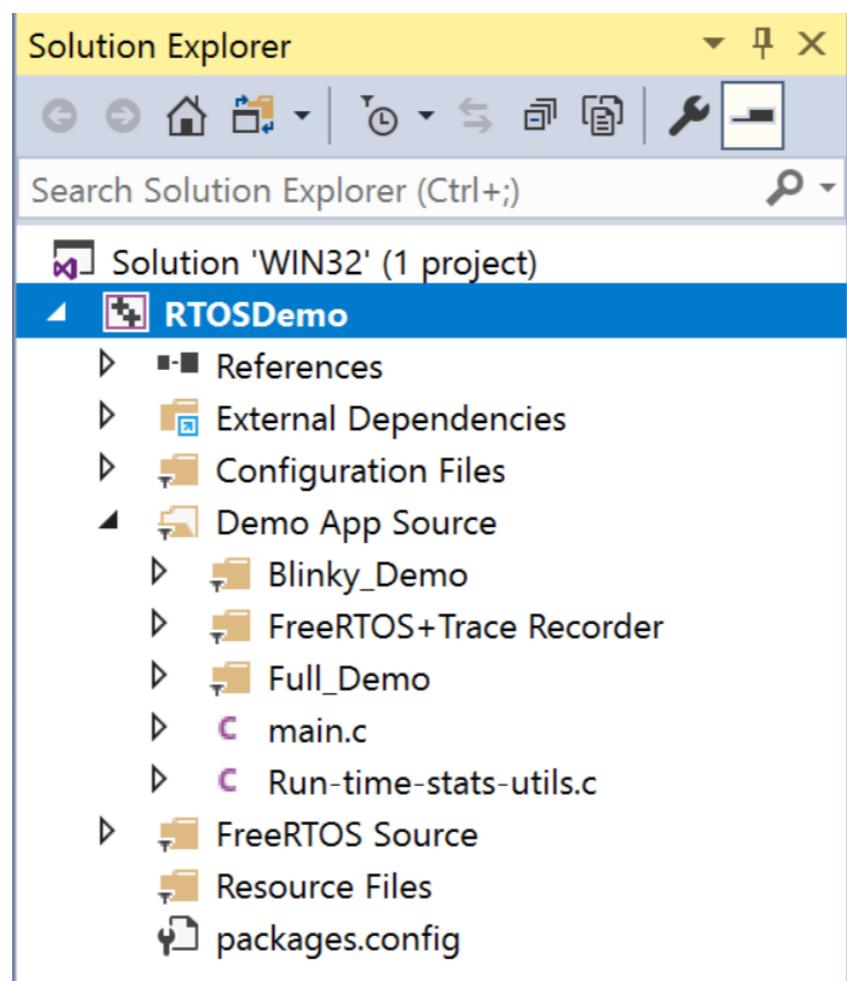
Step 17c. Results



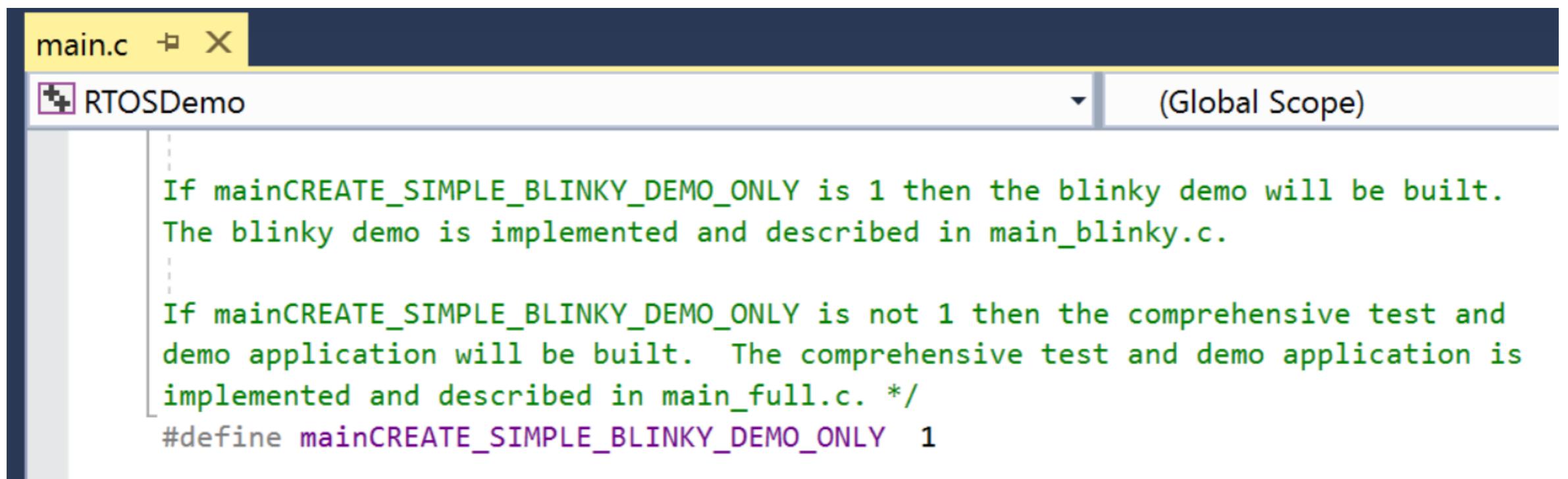
The screenshot shows the 'Output' tab in the Visual Studio IDE. The status bar at the bottom indicates 'Retargeting End: 1 completed, 0 failed, 0 skipped'. The main area displays the following text:

```
Upgrading project 'RTOSDemo'...
Configuration 'Debug|Win32': changing Platform Toolset to 'v141' (was 'v140').
Retargeting End: 1 completed, 0 failed, 0 skipped
```

Step 18. Look at Solution Explorer



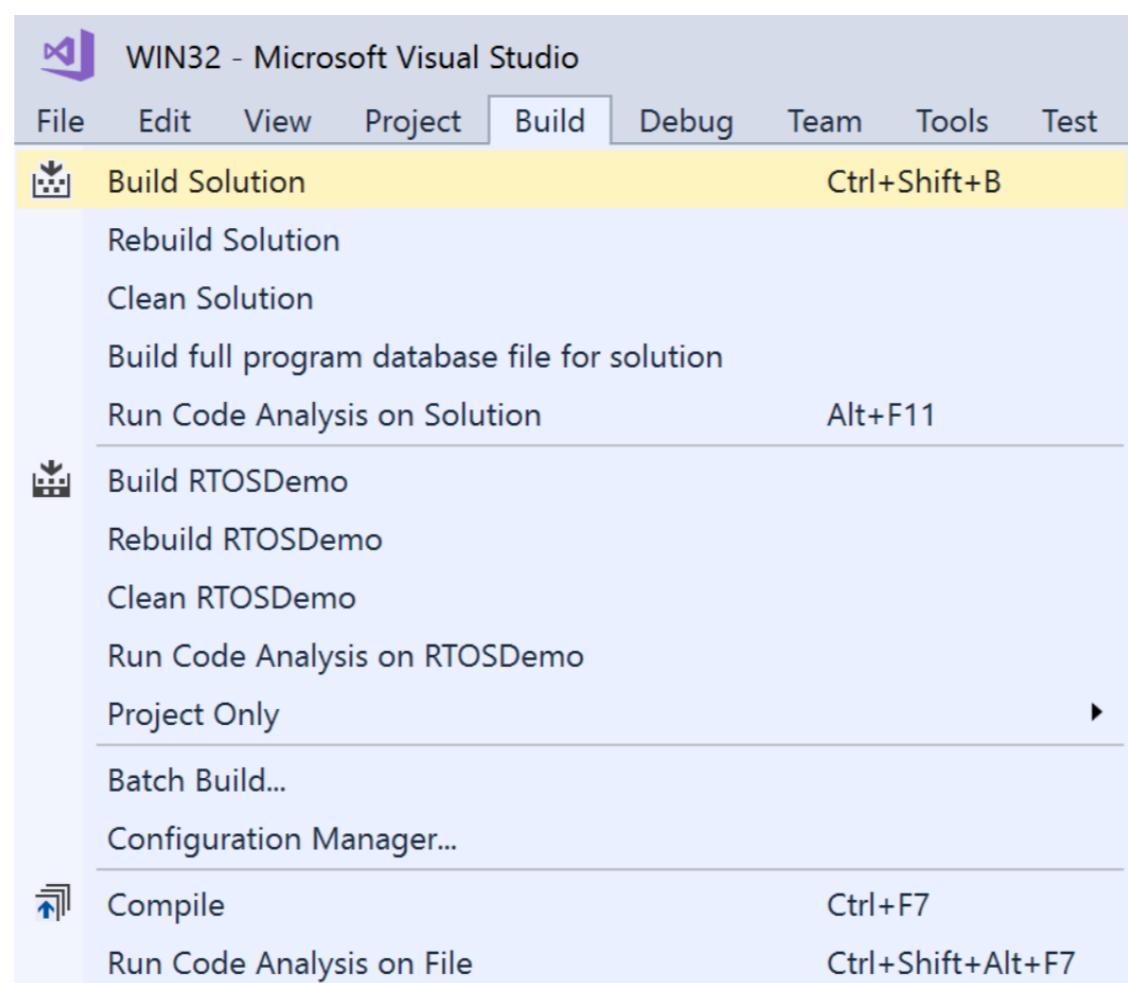
Step 19. Look at main.c, and make sure
mainCREATE_SIMPLE_BLINKY_DEMO_ONLY is set to 1



The screenshot shows a code editor window titled "main.c" with the file content displayed. The file contains C code with some explanatory text. The code is as follows:

```
If mainCREATE_SIMPLE_BLINKY_DEMO_ONLY is 1 then the blinky demo will be built.  
The blinky demo is implemented and described in main_blinky.c.  
  
If mainCREATE_SIMPLE_BLINKY_DEMO_ONLY is not 1 then the comprehensive test and  
demo application will be built. The comprehensive test and demo application is  
implemented and described in main_full.c. */  
#define mainCREATE_SIMPLE_BLINKY_DEMO_ONLY 1
```

Step 18a. Build Solution



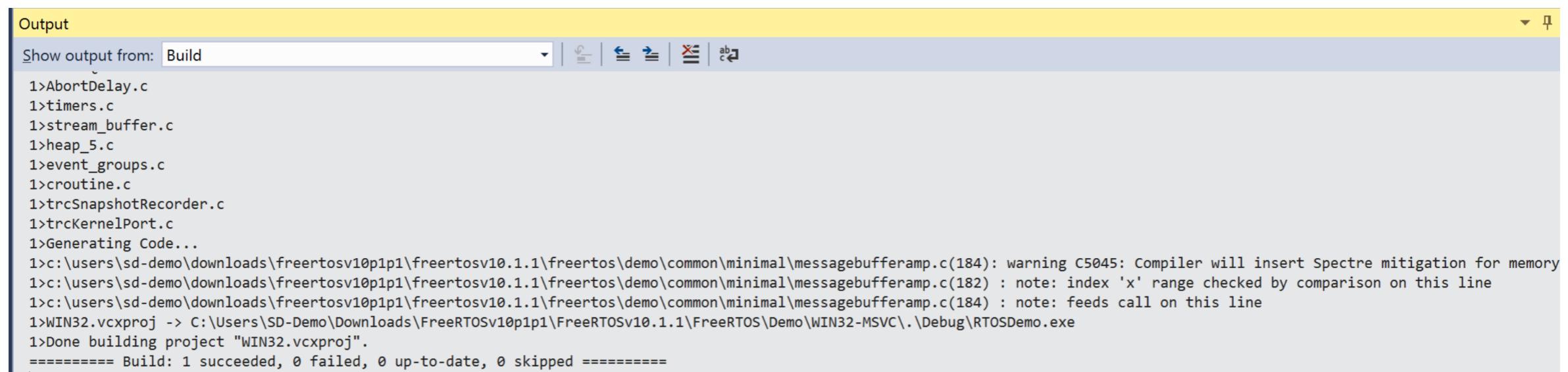
Step 18b. Build in progress

Output

Show output from: Build

```
1>cl : Command line warning D9002: ignoring unknown option '/wc4574'  
1>Run-time-stats-utils.c  
1>main_full.c  
1>main_blinky.c  
1>tasks.c  
1>queue.c  
1>port.c  
1>list.c  
1>main.c  
1>timerdemo.c  
1>TaskNotify.c  
1>StreamBufferInterrupt.c  
1>StreamBufferDemo.c  
1>MessageBufferDemo.c  
1>StaticAllocation.c
```

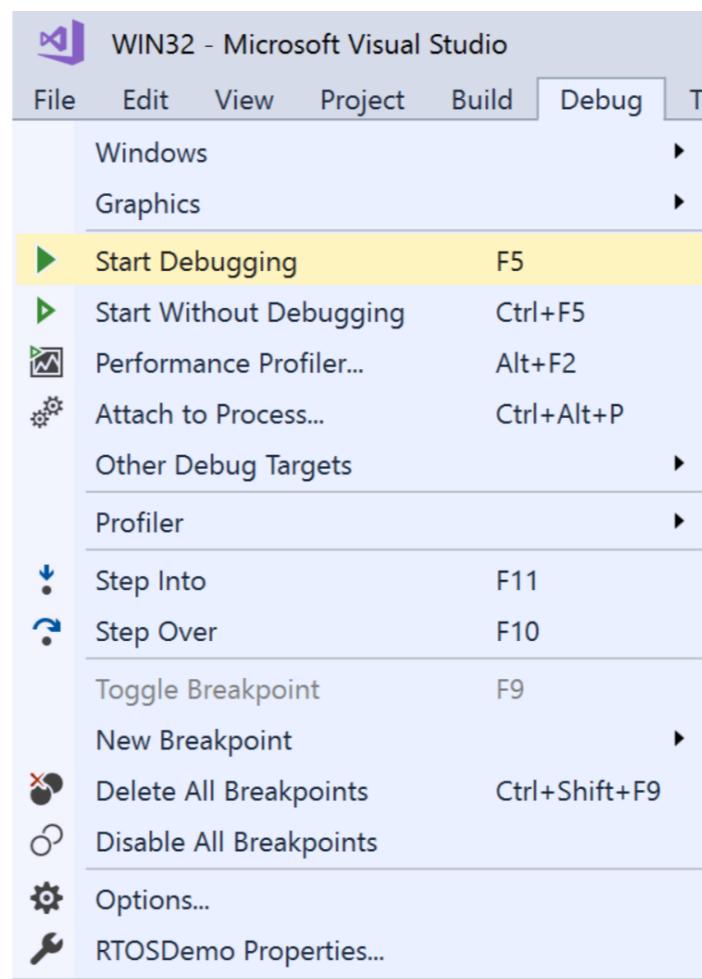
Step 18c. Build Complete



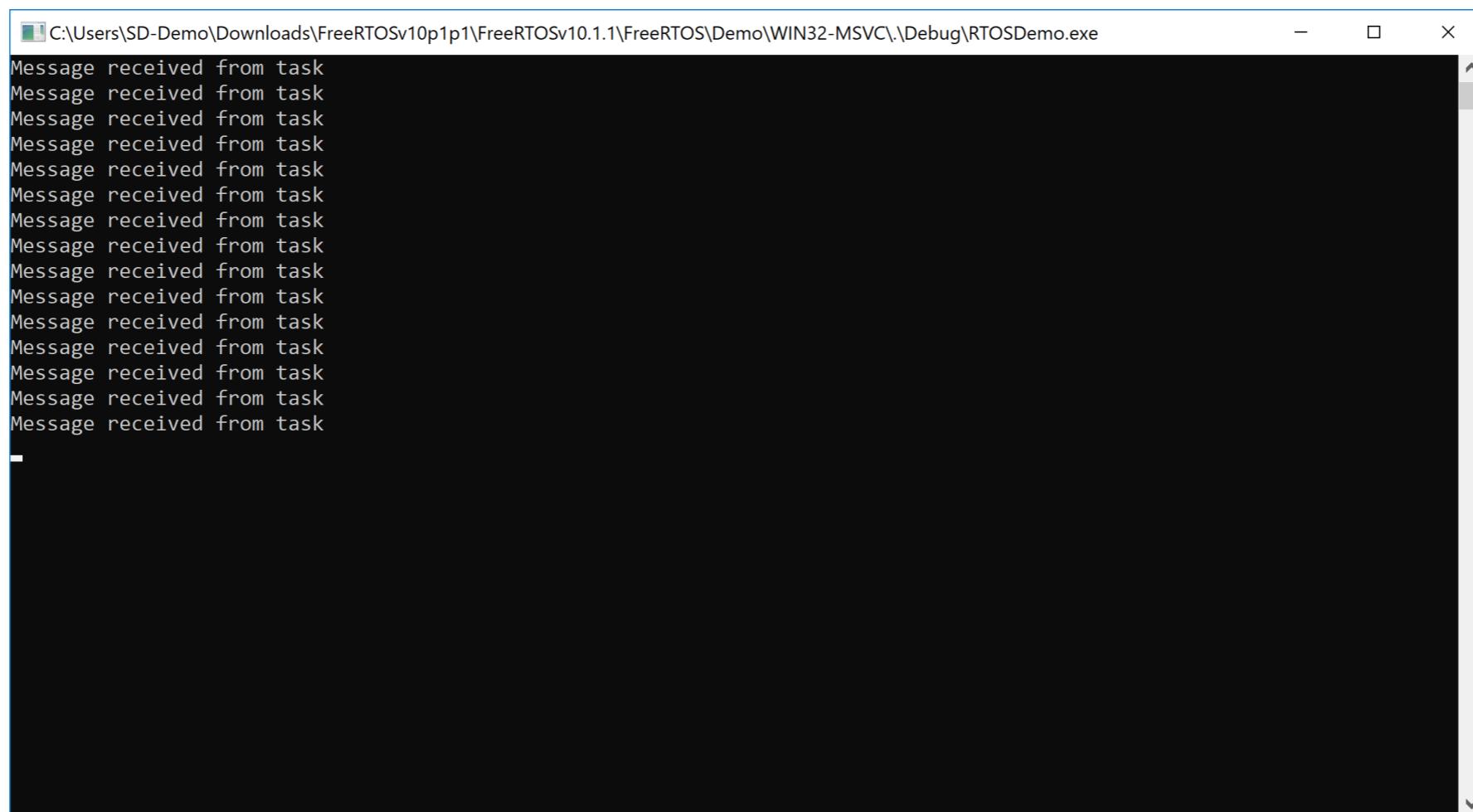
The screenshot shows the 'Output' window from Microsoft Visual Studio. The title bar says 'Output'. The dropdown menu shows 'Build' is selected. Below the menu are several icons: a dropdown arrow, a magnifying glass, a double left arrow, a double right arrow, a red X, and a refresh symbol. The main text area contains the following build log:

```
1>AbortDelay.c
1>timers.c
1>stream_buffer.c
1>heap_5.c
1>event_groups.c
1>croutine.c
1>trcSnapshotRecorder.c
1>trcKernelPort.c
1>Generating Code...
1>c:\users\sd-demo\downloads\freertosv10p1p1\freertosv10.1.1\freertos\demo\common\minimal\messagebufferamp.c(184): warning C5045: Compiler will insert Spectre mitigation for memory
1>c:\users\sd-demo\downloads\freertosv10p1p1\freertosv10.1.1\freertos\demo\common\minimal\messagebufferamp.c(182) : note: index 'x' range checked by comparison on this line
1>c:\users\sd-demo\downloads\freertosv10p1p1\freertosv10.1.1\freertos\demo\common\minimal\messagebufferamp.c(184) : note: feeds call on this line
1>WIN32.vcxproj -> C:\Users\SD-Demo\Downloads\FreeRTOSv10p1p1\FreeRTOSv10.1.1\FreeRTOS\Demo\WIN32-MSVC\.\Debug\RTOSDemo.exe
1>Done building project "WIN32.vcxproj".
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ======
```

Step 19. Start Debugging



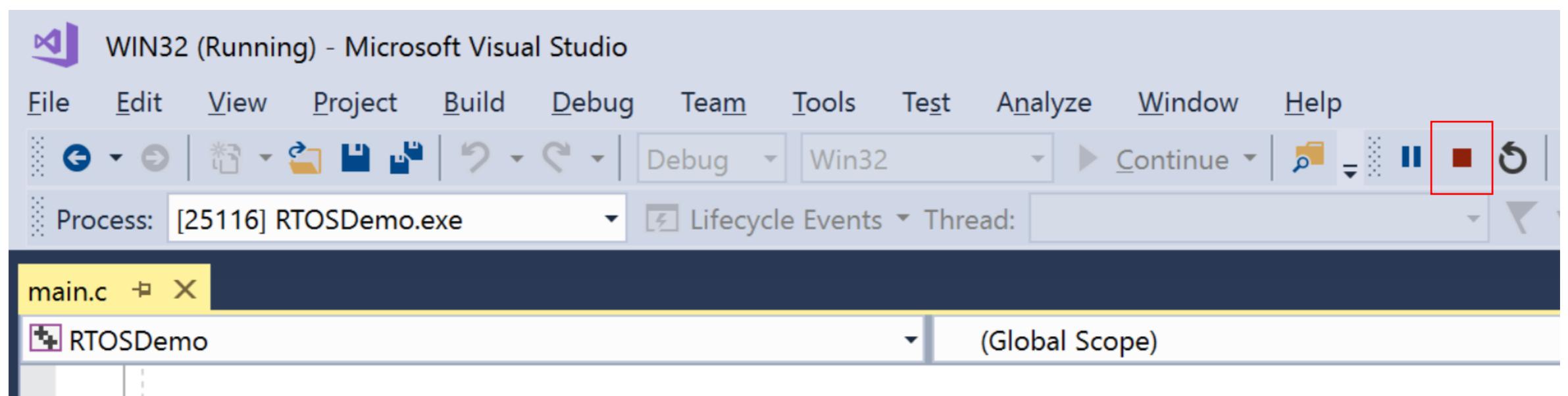
Step 20. Running!



A screenshot of a terminal window titled "C:\Users\SD-Demo\Downloads\FreeRTOSv10p1p1\FreeRTOSv10.1.1\FreeRTOS\Demo\WIN32-MSVC.\Debug\RTOSDemo.exe". The window has a black background and contains white text. It displays a series of messages: "Message received from task" repeated 15 times, followed by a single hyphen '-'.

```
Message received from task
-
-
```

Step 21. Stop Debugging



Summary

- You now have the skills to install and start running the FreeRTOS Windows Simulator