



Debugging Hello World in C on a Fixed Virtual Platform

Version 1.0

Non-Confidential

Copyright © 2020 Arm Limited (or its affiliates).
All rights reserved.

Issue 00

102628_0100_00_en



Debugging Hello World in C on a Fixed Virtual Platform

Copyright © 2020 Arm Limited (or its affiliates). All rights reserved.

Release information

Document history

Issue	Date	Confidentiality	Change
0100-00	1 January 2020	Non-Confidential	Initial release

Proprietary Notice

This document is protected by copyright and other related rights and the practice or implementation of the information contained in this document may be protected by one or more patents or pending patent applications. No part of this document may be reproduced in any form by any means without the express prior written permission of Arm. No license, express or implied, by estoppel or otherwise to any intellectual property rights is granted by this document unless specifically stated.

Your access to the information in this document is conditional upon your acceptance that you will not use or permit others to use the information for the purposes of determining whether implementations infringe any third party patents.

THIS DOCUMENT IS PROVIDED "AS IS". ARM PROVIDES NO REPRESENTATIONS AND NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, NON-INFRINGEMENT OR FITNESS FOR A PARTICULAR PURPOSE WITH RESPECT TO THE DOCUMENT. For the avoidance of doubt, Arm makes no representation with respect to, and has undertaken no analysis to identify or understand the scope and content of, patents, copyrights, trade secrets, or other rights.

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly

or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

The Arm corporate logo and words marked with ® or ™ are registered trademarks or trademarks of Arm Limited (or its affiliates) in the US and/or elsewhere. All rights reserved. Other brands and names mentioned in this document may be the trademarks of their respective owners. Please follow Arm’s trademark usage guidelines at <https://www.arm.com/company/policies/trademarks>.

Copyright © 2020 Arm Limited (or its affiliates). All rights reserved.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349|version 21.0)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Feedback

Arm® welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on <https://support.developer.arm.com>

To provide feedback on the document, fill the following survey: <https://developer.arm.com/documentation-feedback-survey>.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

We believe that this document contains no offensive language. To report offensive language in this document, email terms@arm.com.

Contents

1. Introduction.....	6
2. Create a DS-5 debug configuration and connect to an FVP.....	7
3. Step through the application.....	15
4. Disconnect from the debug connection.....	16

1. Introduction

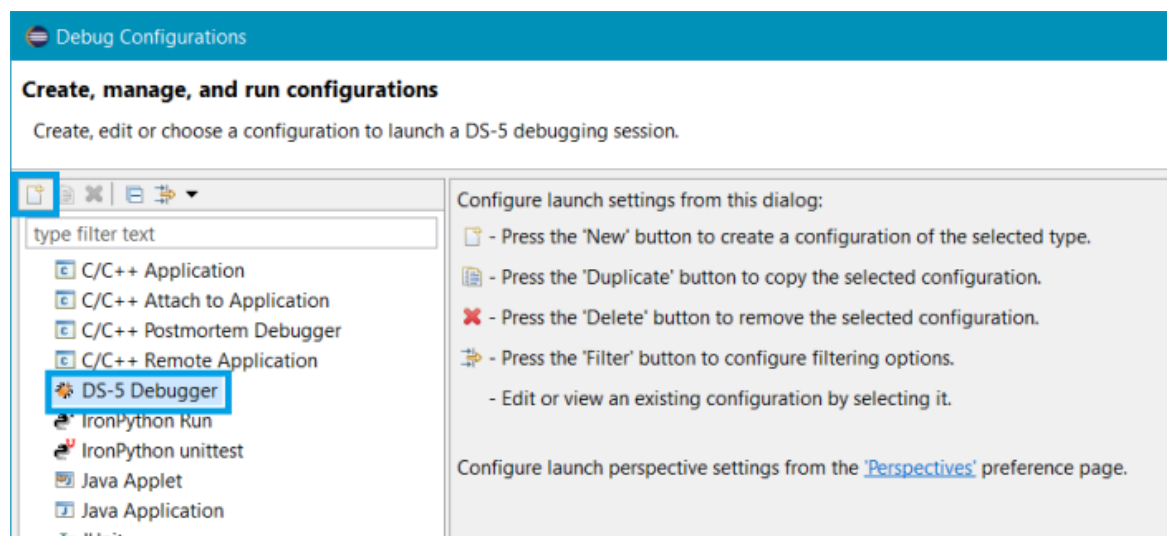
Once you have created the project and built the code, launch the debugger to run the application on one of the Fixed Virtual Platforms (FVP) provided with DS-5.

For this tutorial, we use a Cortex-A9 Fixed Virtual Platform (FVP) which is provided with DS-5.

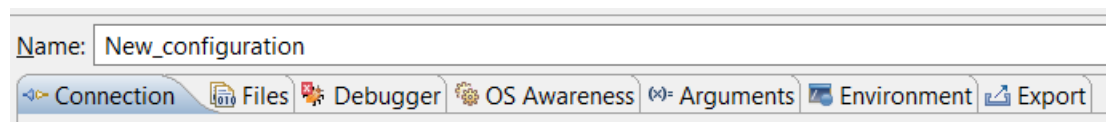
2. Create a DS-5 debug configuration and connect to an FVP

The following steps help you create a DS-5 debug configuration and connecting to an FVP.

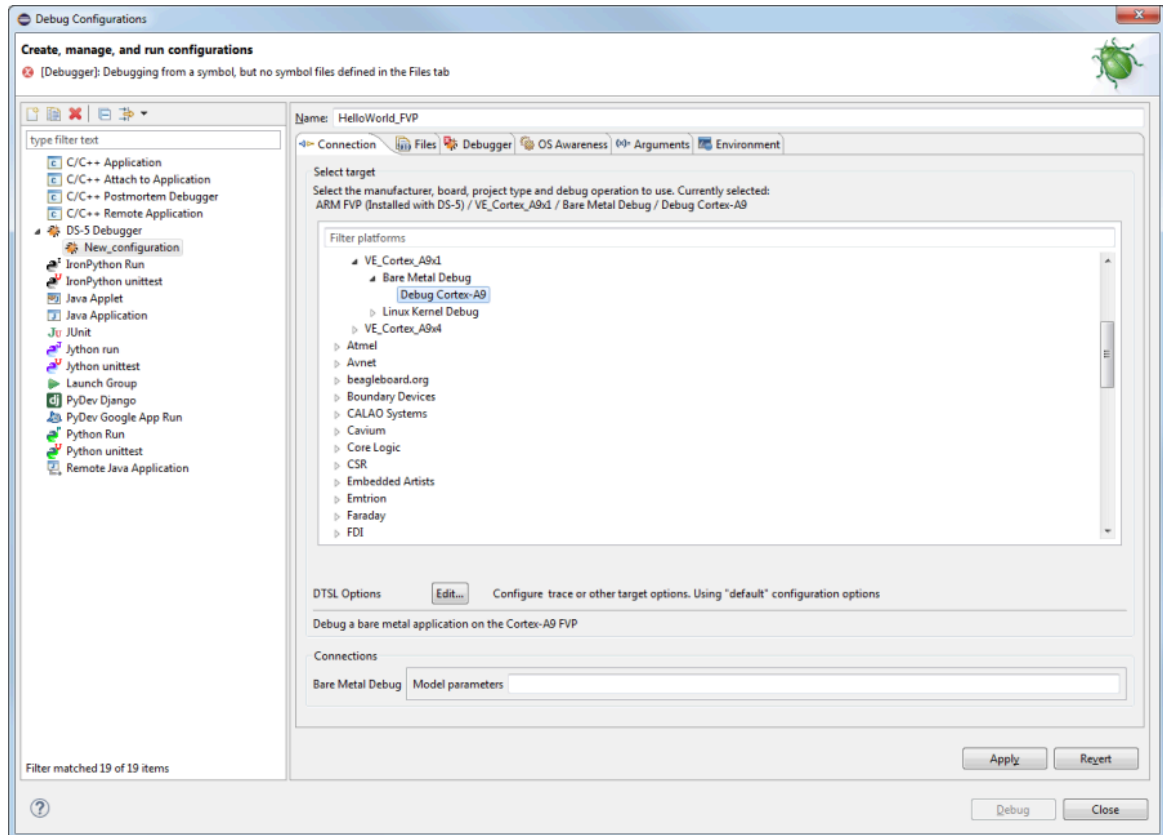
1. From the DS-5 main menu, select Run > Debug Configurations.
2. In the Debug Configurations dialog:
 - a. Select DS-5 Debugger.
 - b. Click the New launch configuration button.



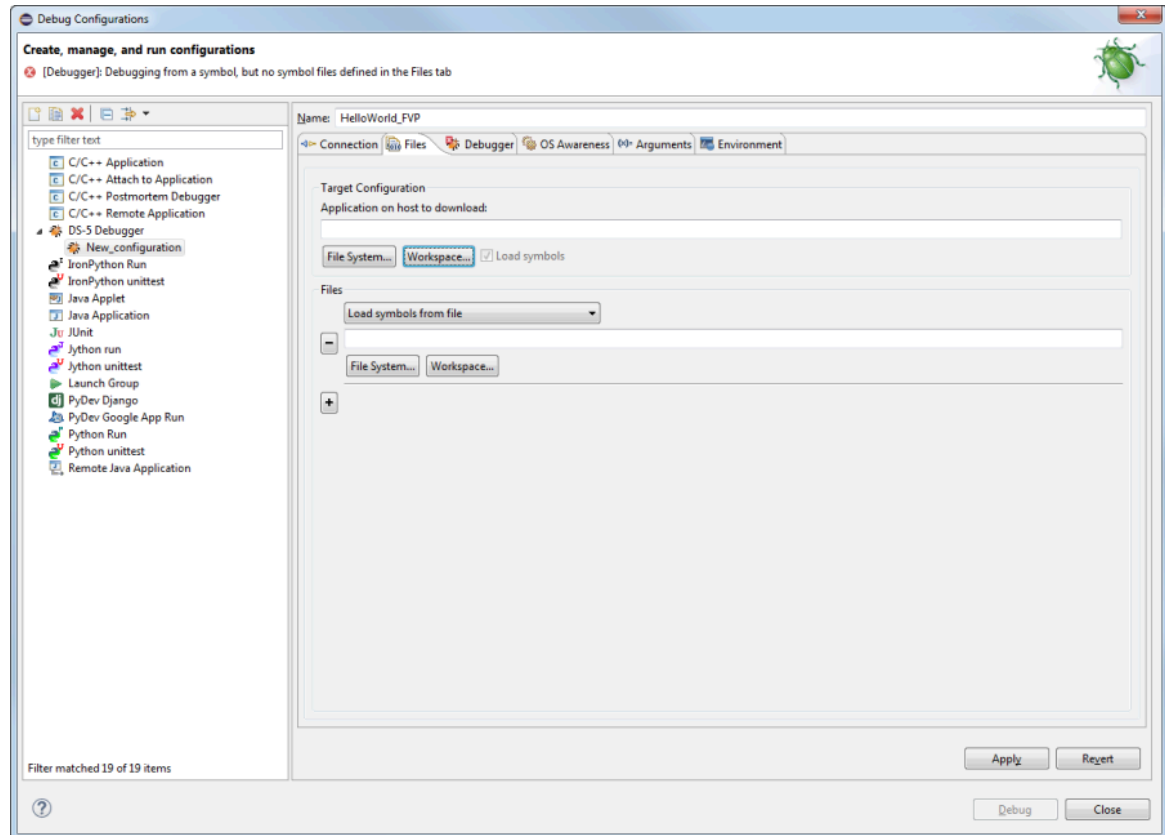
This creates a new DS-5 debug configuration and displays the various tabs required to specify settings for loading your application on the target.



3. In the Debug Configurations dialog:
 - a. Give a name to the debug configuration. For example, HelloWorld_FVP.
 - b. In the Connection tab, under Select Target, browse and select ARM FVP (Installed with DS-5) > VE_Cortex_A9x1 > Bare Metal Debug > Debug Cortex-A9.



- c. Select the Files tab, and under Target Configuration in the Application on host to download field, click Workspace.



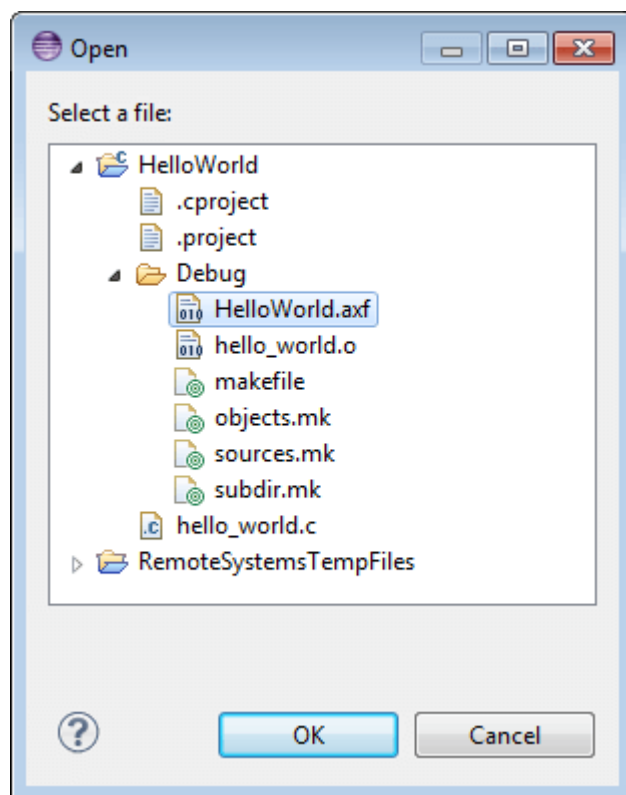
The Workspace contains the `HelloWorld.axf` application file you created when you built the Hello World project.



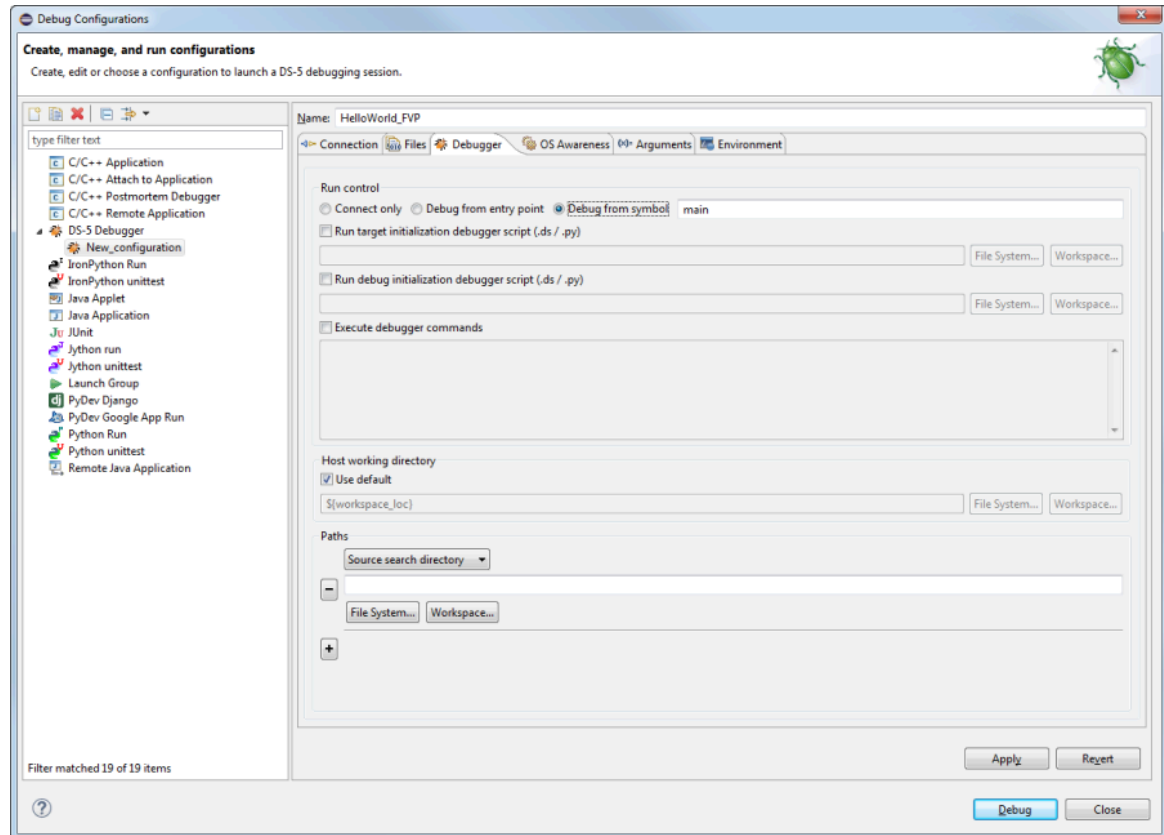
Note

Ensure that the Load symbols option is selected.

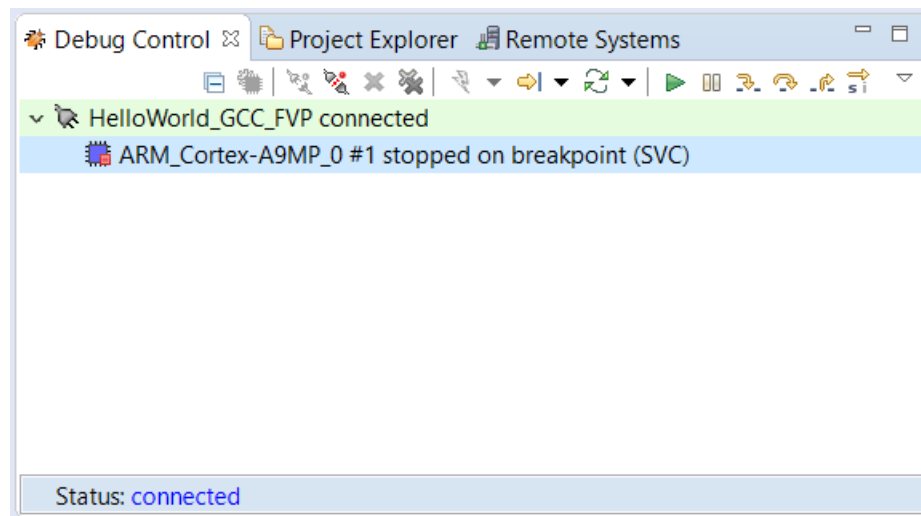
- d. Select `HelloWorld.axf`.



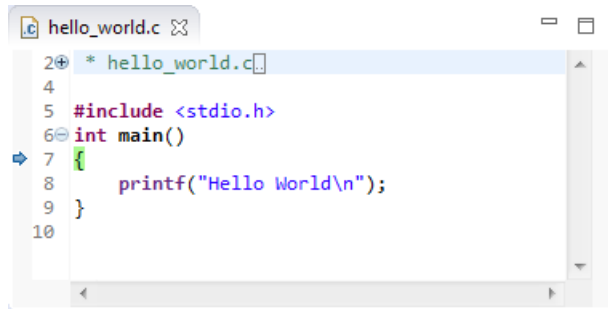
- e. Select the Debugger tab, and ensure the Debug from symbol option is selected and set to main.




- f. Click Debug to load the application on the target, and load the debug information into the debugger.
- g. In the Confirm Perspective Switch dialog that appears, click Yes. DS-5 connects to the model and displays the connection status in the Debug Control view.

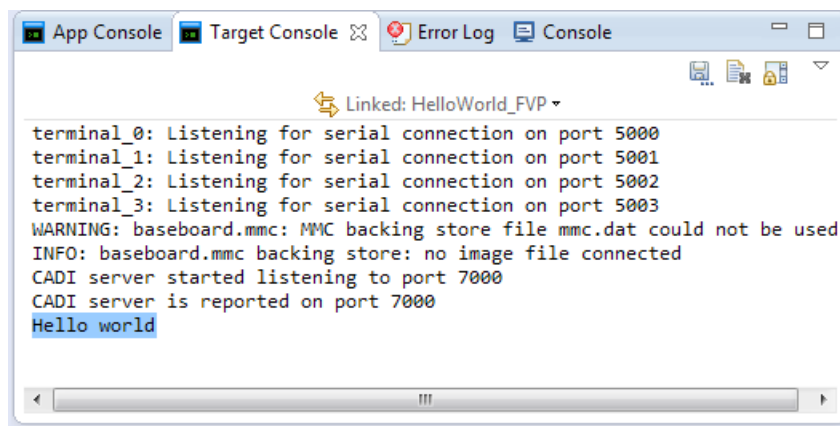


The application is loaded on the target, and has stopped at the `main()` function, ready to run.



```
1  * hello_world.c
2
3
4
5  #include <stdio.h>
6
7  int main()
8  {
9      printf("Hello World\n");
10 }
```

- h. Click  to continue running the application. You can view the application output in the Target Console view.



```
App Console | Target Console | Error Log | Console
Linked: HelloWorld_FVP
terminal_0: Listening for serial connection on port 5000
terminal_1: Listening for serial connection on port 5001
terminal_2: Listening for serial connection on port 5002
terminal_3: Listening for serial connection on port 5003
WARNING: baseboard.mmc: MMC backing store file mmc.dat could not be used
INFO: baseboard.mmc backing store: no image file connected
CADI server started listening to port 7000
CADI server is reported on port 7000
Hello world
```

Other views display information relevant to the debug connection

- Commands view displays messages output by the debugger. Also use this view to enter DS-5 commands.

```

Commands History Scripts
Linked: HelloWorld_FVP
Connected to stopped target ARM FVP (Installed with DS-5) - VE_Cortex_A9x1
Execution stopped at: S:0x00000000
loadfile "C:\DS-5_Workspace\Hello World\Debug\Hello World.axf"
S:0x00000000 DCI 0xe7ff0010 ; ? Undefined
Loaded section ER_RO: S:0x80000000 ~ S:0x800013AB (size 0x13AC)
Loaded section ER_RW: S:0x800013AC ~ S:0x800013BB (size 0x10)
Entry point S:0x80000000
cd "C:\DS-5_Workspace"
Semihosting server socket created at port 8000
Semihosting enabled automatically due to semihosting symbol detected in image 'Hello World.axf'
Working directory "C:\DS-5_Workspace"
set debug-from main
start
Starting target with image C:\DS-5_Workspace\Hello World\Debug\Hello World.axf
Running from entry point
Execution stopped at breakpoint 1: S:0x800000B4
In hello_world.c
S:0x800000B4 7,1 {
Deleted temporary breakpoint: 1
wait
wait
continue
Execution stopped at: S:0x80001124
In _sys_exit (no debug info)
S:0x80001124 SVC #0x123456

Command: Press (Ctrl+Space) for Content Assist Submit

```

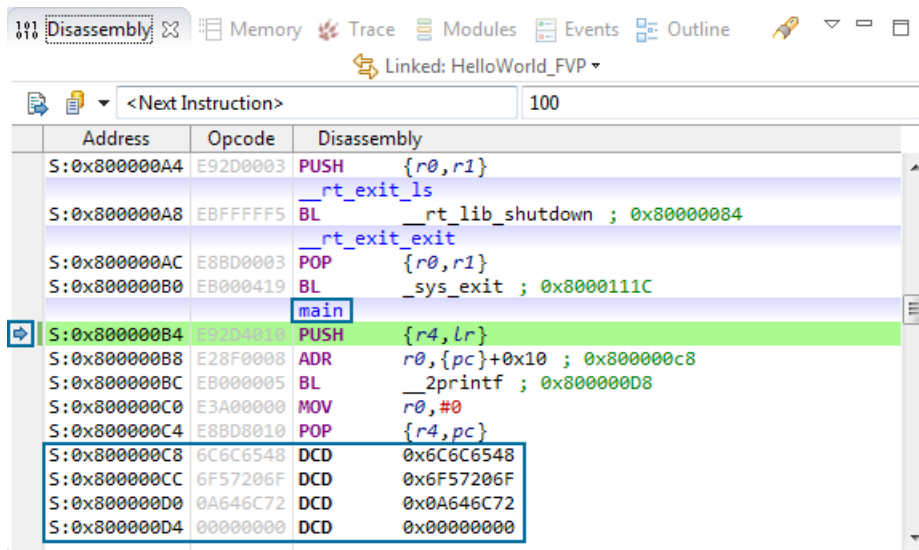
- C/C++ Editor view shows the structure of the active C, C++, or makefile. The view is updated as you edit these files.

```

hello_world.c
2+ * hello_world.c
4
5 #include <stdio.h>
6 int main()
7 {
8     printf("Hello World\n");
9 }
10

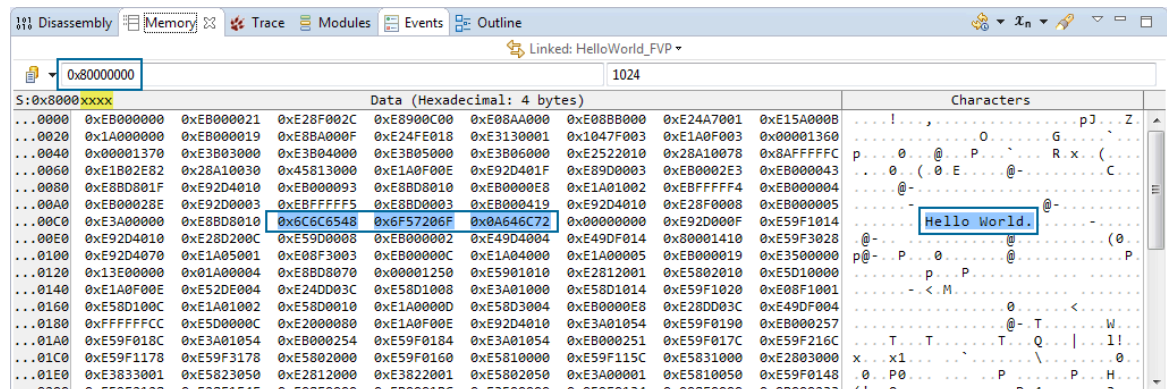
```

- Disassembly view shows the loaded program in memory as addresses and assembler instructions.



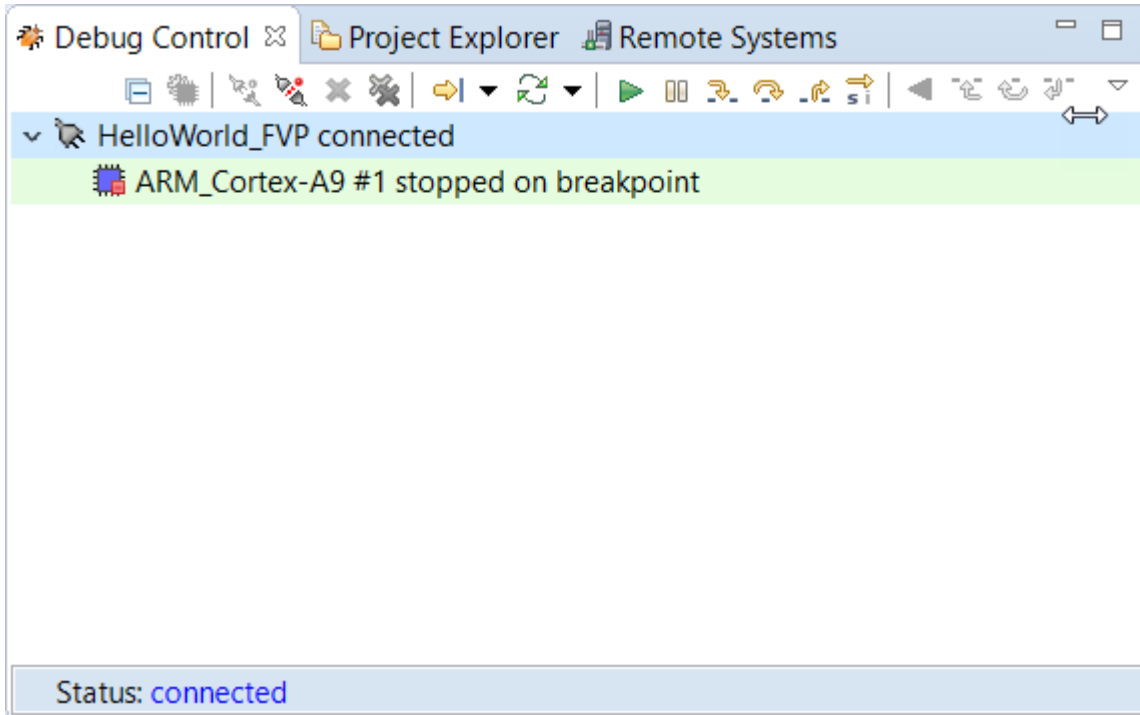
Indicates the location in the code where your program is stopped. In this case, it is at the `main()` function.

- Memory view shows how the code is represented in the target memory. For example, to view how the string `Hello World` from the application is represented in memory:
 - Open the Memory view.
 - In the Address field, enter `0x80000000` and press Enter on your keyboard. The view displays contents of the target's memory.
 - Select and highlight the words `Hello World`.



3. Step through the application

Use the controls provided in the Debug Control view to step through the application.



Click to continue processing code.



Click to interrupt or pause processing code.



Click to step through the code.



Click to step over source line.



Click to step out.



This is a toggle. Select this if you want the above controls to step through instructions.

4. Disconnect from the debug connection

To disconnect from a debug connection, you can either right-click the connection and select Disconnect from Target or select the connection and in the Debug Control view toolbar click 