

### **Getting Started with DS-5 Development Studio**

Version 1.0

### **Tutorial**

Non-Confidential

Copyright  $\ensuremath{\mathbb{C}}$  2021 Arm Limited (or its affiliates). All rights reserved.

**Issue 03** 102702\_0100\_03\_en



### Getting Started with DS-5 Development Studio Tutorial

Copyright © 2021 Arm Limited (or its affiliates). All rights reserved.

### **Release information**

### Document history

Issue	Date	Confidentiality	Change
0100-03	10 January 2021	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, has undertaken no analysis to identify or understand the scope and content of, third party 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 <sup>™</sup> are registered trademarks or trademarks of Arm Limited (or its subsidiaries) 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 © 2021 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)

### **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<sup>®</sup> 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. Overview	6
2. Downloading and Installing DS-5	7
3. Starting DS-5 and Setting up your Workspace	8
4. Obtaining a License	9
5. Creating a simple Hello World C program for a bare-metal target	
6. Creating a New C Project	17
7. Specifying a RAM Base Address	19
8. Building the Project	21
9. Debug the application on a Fixed Virtual Platform	
10. Create a DS-5 Debug Configuration and Connecting to an FVP	23
11. Step Through the Application	32
12. Disconnect from the Debug Connection	

## 1. Overview

This tutorial takes you through the process of downloading and installing the evaluation version of Arm DS-5 Development Studio. It then guides you through creating a bare-metal Hello World application and finally running it on a debug configuration for a Cortex-A9 Fixed Virtual Platform (FVP) provided with DS- 5.

## 2. Downloading and Installing DS-5

DS-5 is available for both Windows and Linux hosts. See DS-5 System Requirements for a list of supported hosts.

Download the appropriate DS-5 installer for your host (either Windows or Linux) here.

### Windows Installation Instructions

Extract the files from the downloaded .zip file and run setup.exe. Then follow the simple on-screen installation instructions.



During installation, you might receive warnings such as "Windows can't verify the publisher of this driver software" you can safely ignore these warnings and continue with the installation.

When installed, on Windows 7 platforms, you can find DS-5 under Start menu > All Programs > Eclipse for DS-5

### Linux Installation Instructions

Extract the installer from the downloaded archive file, run (not source) install.sh and follow the on-screen instructions. The installer unpacks DS-5 into your chosen directory, and optionally installs device drivers and desktop shortcuts.



The installer includes device drivers that require you to run with root privileges.

# 3. Starting DS-5 and Setting up your Workspace

To start DS-5 and set up your workspace, do the following:

- 1. To start DS-5, from your desktop menu, select Eclipse for DS-5.
- 2. In the Workspace Launcher dialog, either accept the default workspace, or click Browse and select a folder. For example, C:\_Workspace.

#### Figure 3-1: A description of the image for screen readers.

Geworkspace Launcher				
Select a workspace				
Eclipse Platform stores Choose a workspace for	s your projects in a folder called a workspace. older to use for this session.			
Workspace: C:\DS-5	Vorkspace	~	<u>B</u> rowse	
$\mathbf{\nabla}$ Use this as the defa	ult and do not ask again			
		ОК	Cancel	

- 3. Select Use this as the default and do not ask again option and click OK.
- 4. If you are using DS-5 for the first time, then the No License Found dialog is displayed.
- 5. Click Open License Manager to use the License Manager to add a license.

## 4. Obtaining a License

After installing DS-5, you have to acquire a license to use it. For this tutorial, we are going to use a 30-day evaluation license that allows you to use DS-5 Ultimate Edition for 30 days without any restrictions.



You need to be connected to the internet and have an Arm developer (Silver) account to obtain a DS-5 evaluation license.

1. In the No Licenses Found dialog, click Open License Manager.

### Figure 4-1: A description of the image for screen readers.



2. In the Arm License Manager dialog, click Add License....

### Figure 4-2: A description of the image for screen readers.

ARM Licer	nse Manage	r	×
View and ed Configure lic	dit license enses and c	<b>s</b> liagnose licensing problems.	
Configuration	Diagnost	ics	
License	e Origin		Add License
		No installed licenses found.	Delete License
Select the to	olkit that y	ou intend to use:	
No toolkits	available		$\sim$
?			Close

3. In the Add License dialog, select the Generate a 30-day evaluation license for DS-5 Ultimate Edition option and click Next.

### Figure 4-3: A description of the image for screen readers.

Add License			_	_		×
Select License Type						
Select the type of licens	e that you would	like to use.				
O Use a license file, licen	nse server, serial r aluation license fo	number, or activatio or DS-5 Ultimate Ec	n code lition			
?	< Back	Next >	Finish		Cancel	

4. In the Choose network interface dialog, select a network interface.



Arm recommends selecting a physical network interface. If a virtual interface is selected, the license stops working if the MAC address of the interface is changed.

### Figure 4-4: A description of the image for screen readers.

🖨 Add License			_		×
Choose network	interface				
Choose a network	interface that the licer	nse will be locked to	)		
The new license will select a physical net selected and its MA	l be locked to one of y twork interface. The lic C address later change	vour computer's net cense will stop work es.	work interfaces. It is r ing if a virtual netwo	ecommend rk interface	ed to is
Network interface:	C85B76CECBED - Inte	I(R) Ethernet Conne	ction I219-LM		$\sim$
?	< Back	Next >	Finish	Cance	:I

5. In the Developer account details dialog, enter your Arm developer (Silver) account details. If you do not have an account, then create one.

### Figure 4-5: A description of the image for screen readers.

Add License				_		×
Developer account d	letails					
Enter the ARM develop	er (Silver) accoun	t details				
Enter account details:						
Email:						
Password:						
Forgot password? Click	<u>nere</u> to reset your	password.				
Don't have an account?	Click <u>here</u> to creat	te one.				
?	< Back	Next >	Finish		Cancel	

6. Click Finish. When the license is successfully registered, the 30-day evaluation license and the toolkit that is available to you are visible in the Arm License Manager.

### Figure 4-6: A description of the image for screen readers.

ARM License Manager				×
View and edit licenses Configure licenses and diagnos	e licensing problems			
Configuration Diagnostics				
License		Origin		Add License
✓ DS-5 Ultimate Edition (30	-day Evaluation).lic	File from directory	"C:\Users	Delete License
<			>	
Select the toolkit that you inter	d to use:			
DS-5 Ultimate Edition (Evaluat	ion)			~
?				Close

- 7. Close the Arm License Manager and in the Confirm Restart Eclipse dialog, click Restart Eclipse. This restarts Eclipse and brings up the DS-5 Welcome page.
- 8. Either close the DS-5 Welcome page, or click Go to workbench to view the full DS-5 IDE.

### Figure 4-7: A description of the image for screen readers.



You are now ready to use DS-5.

# 5. Creating a simple Hello World C program for a bare-metal target

After installing DS-5 and obtaining a license, this tutorial then takes you through creating, configuring, and building a simple bare-metal program.

To run your application when it is built, this tutorial then takes you through the steps of configuring a debug connection to a system model implemented in software. These models are called Fixed Virtual Platforms (FVP) and some are provided with DS-5. This tutorial uses the ve\_cortex\_A9x1 FVP model which is based on the Cortex-A9 processor.

## 6. Creating a New C Project

To creating a new C project, do the following:

- 1. From the DS-5 main menu, select File > New > C Project to display the C Project dialog.
- 2. In the C Project dialog, in the Project name field, enter HelloWorld as the name of your project.
- 3. Under Project type, select Executable > Hello World ANSI C Project.
- 4. Under Toolchains, select Arm Compiler 6.

### Figure 6-1: A screenshot showing the C Project dialog.

C Project	ЦХ
C Project Create C project of selected type	
Project name       HelloWorld         Image: Second system       Use default location         Location:       C:\DS-5 Workspace\HelloWorld         Choose file system:       default	Browse
Project type: Toolchains:	
<ul> <li>Executable</li> <li>Empty Project</li> <li>Hello World ANSI C Project</li> <li>Shared Library</li> <li>Static Library</li> <li>Makefile project</li> </ul>	5 built-in)
O Kack Next > Finish	Cancel

Learn more about the Arm Compiler toolchain.

Copyright  $\ensuremath{\mathbb{C}}$  2021 Arm Limited (or its affiliates). All rights reserved. Non-Confidential

5. Click Finish to create a C project called Hello World. You can view the project in the Project Explorer view.

### Figure 6-2: A screenshot showing the Project Explorer directory.



## 7. Specifying a RAM Base Address

To load and execute the application on the target, before compiling the application, we need to tell the linker the target RAM base address. This ensures that the application is built correctly for the particular target.

The VE global model memory map contains the memory address details required for the VE FVP model used in this tutorial.

We can see that the memory address range for VE FVP models (4GB DRAM (in 32-bit address space)) is between  $0 \times 80000000$  and  $0 \times FFFFFFFF$ . This gives us the RAM base address as  $0 \times 80000000$ .

- 1. In Project Explorer, right-click the project and select Properties.
- 2. In the Properties dialog, browse to C/C++ Build > Settings.
- 3. Under the Tool Settings tab, browse to All Tools Settings > Target, and from the Target CPU drop down select **Cortex-A9**.
- 4. Under the Tool Settings tab, browse to All Tools Settings > Target, and from the Target FPU drop down select **no FPU**.
- 5. Under the Tool Settings tab, browse to Arm Linker 6 > Image Layout.
- 6. In the RO base address (-ro\_base) field, enter 0x80000000.

pe filter text	Settings			$\langle \neg \neg \neg \rangle$
pe filter text Resource Builders • C/C++ Build Build Variables Environment Logging Settings Tool Chain Editor C/C++ General Project References Run/Debug Settings	Settings Configuration: Debug [Active] Tool Settings Build Steps Tool Settings Target Debugging Libraries V Starget Preprocessor Includes Source Language Optimizations Debugging Warnings and Errors Miscellaneous V Starget	Build Artifact Binary Par Image entry point (entry) RO base address (ro_base) RW base address (rw_base) ZI base address (zi_base) Scatter file (scatter) Predefine (pd)	rsers S Er       0x8000000	
	<ul> <li>Preprocessor</li> <li>Includes</li> <li>Debugging</li> <li>Warnings and Errors</li> <li>Miscellaneous</li> <li>Miscellaneous</li> <li>Arm Linker 6</li> <li>Target</li> <li>Image Layout</li> <li>Libraries</li> <li>Optimizations</li> <li>Additional Information</li> <li>Warnings and Errors</li> </ul>			

### Figure 7-1: A screenshot showing the RO base settings.

7. Click OK to close the dialog and apply the changes.

## 8. Building the Project

In the Project Explorer view, right-click on the Hello World project and select Build Project.

You can view the output image HelloWorld.axf in the Debug folder under the HelloWorld project.

The .axf file contains both the object code and debug symbols that enable the debugger to perform source-level debugging.

### 9. Debug the application on a Fixed Virtual Platform

When 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.

## 10. Create a DS-5 Debug Configuration and Connecting to an FVP

To create a DS-5 debug configuration and connect to an FVP, do the following.

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

### Figure 10-1: A screenshot showing the DS-5 Debugger menu.

Debug Configurations

### Create, manage, and run configurations

Create, edit or choose a configuration to launch a DS-5 debugging session.

	Configure launch settings from this dialog:
type filter text	Press the 'New' button to create a configuration of the selected type.
C/C++ Application	Press the 'Duplicate' button to copy the selected configuration.
C/C++ Attach to Ap	Proce the 'Delete' butten to remove the collected configuration
C/C++ Postmortem	A - Press the Delete button to remove the selected configuration.
C/C++ Remote App	Press the 'Filter' button to configure filtering options.
🌞 DS-5 Debugger	- Edit or view an existing configuration by selecting it.
IronPython Run	,,,,,,, _

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

Figure 10-2: A screenshot showing the Connection tab.

<u>N</u> ame:	New_configuration
<ul> <li>Cor</li> </ul>	nnection 🛛 🐻 Files 👺 Debugger 🆓 OS Awareness 🕺 Arguments 🚾 Environment

- 4. In the Debug Configurations dialog, give a name to the debug configuration. For example, HelloWorld\_FVP\*.
- 5. 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.

### Figure 10-3: A screenshot showing the Debug Cortex-A9 option.

🗎 🗶 📄 🛟 ▼	Name: HelloWorld_FVP	
pe filter text C/C++ Application C/C++ Attach to Application C/C++ Postmortem Debugge C/C++ Remote Application D-5 Debugger New_configuration C/Log Pure	<ul> <li>Connection Files Debugger OS Awareness Arguments Environment</li> <li>Select target</li> <li>Select the manufacturer, board, project type and debug operation to use. Currently selected: ARM FVP (Installed with DS-5) / VE_Cortex_A9x1 / Bare Metal Debug / Debug Cortex-A9</li> <li>Filter platforms</li> <li>VE_Cortex_A5x1</li> </ul>	
<ul> <li>IronPython Kun</li> <li>IronPython unittest</li> <li>Java Applet</li> <li>Java Application</li> <li>Ju JUnit</li> <li>Jython run</li> <li>Jython unittest</li> </ul>	VE_Cortex_A7x1 VE_Cortex_A9x1 Bare Metal Debug Debug Cortex-A9 Linum Kernel Debug	~
<ul> <li>Launch Group</li> <li>PyDev Django</li> <li>PyDev Google App Run</li> <li>Python Run</li> <li>Python unittest</li> <li>Remote Java Application</li> </ul>	DTSL Options Edit Configure trace or other target options. Using "default" configura Debug a bare metal application on the Cortex-A9 FVP Connections Bare Metal Debug Model parameters	tion options

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

### Figure 10-4: A screenshot showing the Workspace option.

Image: Second	Name: HelloWorld_FVP  Connection Files Debugger OS Awareness In Arguments Environment  Target Configuration Application on host to download:  File System Workspace  Files Files Files File System Workspace  +
< >>	Revert Apply

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



7. Select HelloWorld.axf.

🖨 Open	-		×
Select a file:			
🗸 😂 HelloWorld			
cproject			
📄 .project			
> 🗁 .settings			
🗸 🗁 Debug			
🗟 HelloWorld.axf			
hello_world.d			
hello_world.o			
🚡 makefile			
objects.mk			
là sources.mk			
là subdir.mk			
le hello_world.c			
RemoteSystemsTempFiles			
?	OK	Cancel	

Figure 10-5: A screenshot showing the HelloWorld.axf file location.

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

### Figure 10-6: A screenshot showing the Debug from symbol option.

ĵ 🗎 🗶 🖻 ‡> ▼	Name: HelloWorld_FVP			_
type filter text	← Connection 🔚 Files 🏶 Debugger 🌾 OS Awareness 👐 Arguments 📼	Environment		
C/C++ Application				
C/C++ Attach to Application	Run control			~
C/C++ Postmortem Debugge	○ Connect only ○ Debug from entry point ● Debug from symbol mai	in		
C/C++ Remote Application	Run target initialization debugger script ( ds / pv)			
New configuration		File Custom		
₽ <sup>™</sup> IronPython Run		File System	workspace	
e <sup>v</sup> IronPython unittest	Run debug initialization debugger script (.ds / .py)			
🔊 Java Applet		File System	Workspace	
Java Application	Execute debugger commands			
Ju JUnit			^	
a Jython run				
a Jython unittest				
Launch Group			~	
PyDev Google App Run				
<ul> <li>Python Run</li> </ul>	Host working directory			
Python unittest	Use default			
🖳 Remote Java Application	{workspace_loc}	File System	Workspace	
		Revert	Appl	у

- 9. Click Debug to load the application on the target, and load the debug information into the debugger.
- 10. 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.

🏘 Debug Control 🖾 🍋 Project Explorer 🛛 📕 Remote Systems	- 6	3
📄 🎕   첫 🕺 🗙 🙀   아 ㅋ 운 ㅋ   🕨 🗷 👁 12 중   4 월 20 3		$\overline{}$
✓ 🗽 HelloWorld_FVP connected		
ARM_Cortex-A9 #1 stopped on breakpoint		
Status: connected		

Figure 10-7: A screenshot showing the Debug Control tab.

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

Figure 10-8: A screenshot showing the Hello World function ready to run.



11. Click the green arrow to continue running the application. You can view the application output in the Target Console view.





### 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.

### Figure 10-10: A screenshot showing the Commands tab.



• C/C++ Editor view shows the active C, C++, or Makefile. The view is updated as you edit these files.

•

### Figure 10-11: A screenshot showing the Editor view.



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

101	Disassembly	E Memor	y 🐇 Tra	ace 🗧 Modules 📴 Events 📴 Outline	N	$\bigtriangledown$		
	🔄 Linked: HelloWorld_FVP 🗝							
F	🖇 🗐 🔻 <next ir<="" th=""><th>nstruction&gt;</th><th>100</th><th></th><th></th><th></th></next>	nstruction>	100					
	Address	Opcode	Disass	embly				
	S:0x800000A4	E92D0003	PUSH	{r0,r1}				
			rt_ex	kit_ls				
	S:0x800000A8	EBFFFFF5	BL	rt_lib_shutdown ; 0x80000084				
			rt_ex	kit_exit				
	S:0x800000AC	E8BD0003	POP	{r0,r1}				
	S:0x800000B0	EB000419	BL	_sys_exit ; 0x8000111C				
			main				=	
₽.	S:0x800000B4	E92D4010	PUSH	{r4, lr}				
	S:0x80000088	E28F0008	ADR	r0,{pc}+0x10; 0x800000c8				
	S:0x800000BC	EB000005	BL	2printf ; 0x80000008				
	5:0x800000C0	E3A00000	MOV					
	5:0x000000C4	E0000010	PUP	{r4,pc}				
	5:000000000	65572065	DCD	0x000000040				
	S:0x00000000	01572001	DCD	0x0F57200F				
	S:0x80000000	00000000	DCD	0,00,040072				
	3.070000004	000000000	000	070000000			*	

#### Figure 10-12: A screenshot showing the Disassembly tab.

The blue arrow 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:
- 1. For example, to view how the string Hello World from the application is represented in memory, open the Memory view.
- 2. In the Address field, enter 0x8000000€ and press Enter on your keyboard. The view displays contents of the target's memory.
- 3. Select and highlight the words Hello World.

Figure 10-13:	A screenshot	showing the	Memory tab.
---------------	--------------	-------------	-------------

111 Disas	sembly 🗐 Men	nory 🖾 🐇 Tra	ce 🗧 Module	s 📰 Events 🗄	Outline				& ▼ 𝑥 <sub>n</sub> ▼ 🔗 🗢 🗖
Linked: HelloWorld_FVP ▼									
Ox8000000     1024									
S:0x800	S:0x8000xxxx Data (Hexadecimal: 4 bytes)					Characters			
0000	0×EB000000	0xEB000021	0xE28F002C	0xE8900C00	0xE08AA000	0xE08BB000	0xE24A7001	0xE15A000B	
0020	0x1A000000	0xEB000019	0xE8BA000F	0xE24FE018	0xE3130001	0x1047F003	0xE1A0F003	0x00001360	G`
0040	0x00001370	0xE3B03000	0xE3B04000	0xE3B05000	0xE3B06000	0xE2522010	0x28A10078	0x8AFFFFFC	p0@P` R.x(
0060	0xE1B02E82	0x28A10030	0x45813000	0xE1A0F00E	0xE92D401F	0xE89D0003	0xEB0002E3	0xEB000043	0(.0.E@C
0080	0xE8BD801F	0xE92D4010	0xEB000093	0xE8BD8010	0xEB0000E8	0xE1A01002	0xEBFFFFF4	0xEB000004	
00A0	0xEB00028E	0xE92D0003	0xEBFFFFF5	0xE8BD0003	0xEB000419	0xE92D4010	0xE28F0008	0xEB000005	
00C0	0xE3A00000	0xE8BD8010	0x6C6C6548	0x6F57206F	0x0A646C72	0x00000000	0xE92D000F	0xE59F1014	Hello World.
00E0	0xE92D4010	0xE28D200C	0xE59D0008	0xEB000002	0xE49D4004	0xE49DF014	0x80001410	0xE59F3028	.@(0
0100	0xE92D4070	0xE1A05001	0xE08F3003	0xEB00000C	0xE1A04000	0xE1A00005	0xEB000019	0xE3500000	р@Р0
0120	0x13E00000	0x01A00004	0xE8BD8070	0x00001250	0xE5901010	0xE2812001	0xE5802010	0xE5D10000	pP
0140	0xE1A0F00E	0xE52DE004	0xE24DD03C	0xE58D1008	0xE3A01000	0xE58D1014	0xE59F1020	0xE08F1001	
0160	0xE58D100C	0xE1A01002	0xE58D0010	0xE1A0000D	0xE58D3004	0xEB0000E8	0xE28DD03C	0xE49DF004	
0180	0xFFFFFFCC	0xE5D0000C	0xE2000080	0xE1A0F00E	0xE92D4010	0xE3A01054	0xE59F0190	0xEB000257	W
01A0	0xE59F018C	0xE3A01054	0xEB000254	0xE59F0184	0xE3A01054	0xEB000251	0xE59F017C	0xE59F216C	TTTQ1!
01C0	0xE59F1178	0xE59F3178	0xE5802000	0xE59F0160	0xE5810000	0xE59F115C	0xE5831000	0xE2803000	xx1
01E0	0xE3833001	0xE5823050	0xE2812000	0xE3822001	0xE5802050	0xE3A00001	0xE5810050	0xE59F0148	.0P0
0000	AVEE051110	AUCHOCIC/C	A-COSCOOR	0.00000100	0.00000000	0000000104	A-AAOCAAAAA	0-0000000000	/1 0 0 4 2

In the above example, the Memory view displays the hexadecimal values for the code, and also the ASCII character equivalent of the memory values which enable you to drill down into the details of the code.

## **11.** Step Through the Application

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

### Figure 11-1: A screenshot showing the Debug Control icons.



- Click to continue executing code.
- Click to interrupt or pause executing 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.

## 12. Disconnect from the Debug Connection

To disconnect from a debug connection, you can:

- Right-click the connection and select Disconnect from Target, or
- Select the connection and in the Debug Control view toolbar click <sup> $\times$ </sup>, or
- Double-click on the selected connection.