ADS MultiTrace Tutorial v2.0

Introduction

Aim

This guide provides the user with a basic introduction to the ARM MultiTrace tools and some guidance for their initial setup and use.

The tutorial is split into two sections:

Section 1 – Setting up the MultiTrace unit.
Section 2 – Using MultiTrace with AD.

Pre-requisites

This guide assumes the user has access to a PC workstation with the following tools installed:

ADS 1.2
Multi ICE 2.2
MultiTrace 1.0

The information provided in this guide also applies to later versions of these tools.

Note: Explanation of File Extensions:

`.c`	C source file.
`.h`	C header file.
`.o`	object file.
`.s`	assembly language source file.
`.axf`	ARM Executable file, as produced by **`armlink`**.
`.txt`	ASCII text file.

Additional information

This tutorial is not designed to provide detailed documentation of the MultiTrace tools, as the user guide covers this and is available both on-line and in hard copy. To access the on-line user guide:

From the *Start* menu select *Programs* → *ARM MultiTrace v1.0* → *User Guide*.

Icon conventions

Various icons are used throughout the tutorial to clarify the purpose of text associated with them. Icons either signify the presence of information on a particular topic, or the requirement for user interaction.

The following icons all indicate that user interaction is required:

Indicates other keyboard or mouse input is required

Application icon. Suggests an application to be used to perform a given operation. This example shows *Microsoft Notepad*

Button icon. This indicates that a corresponding button within the current application can be used to perform the operation currently being discussed

The following icons show information:

Suggests that further help is available from other resources.

Identifies a user friendly hint or tip.

Highlights important information regarding the current topic.

Section 1: Setting up the MultiTrace unit

For full, detailed information on this topic please refer to chapter 2 of the MultiTrace user guide.

Requirements include access to a Windows PC with the following:

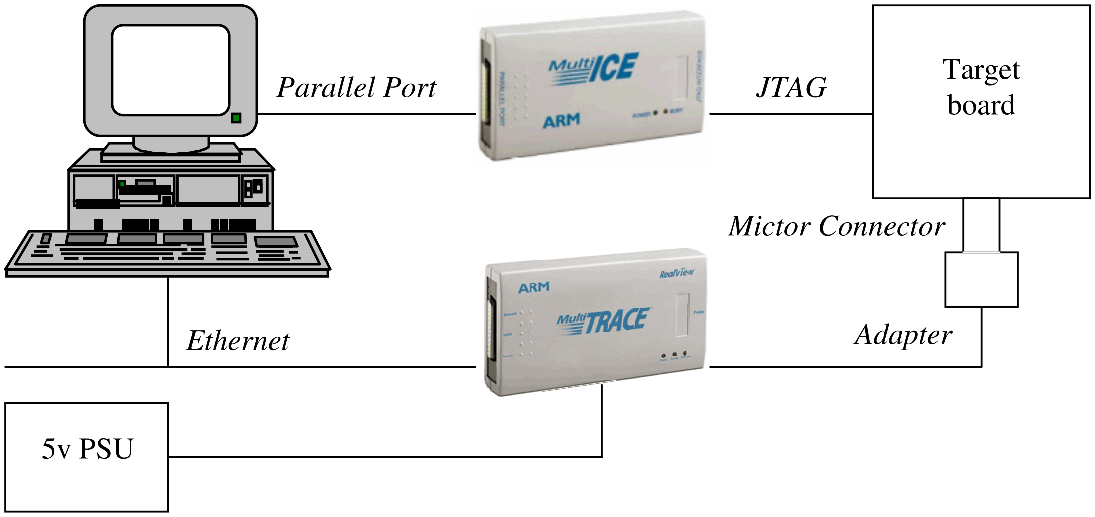
ADS 1.2
Multi-ICE 2.2
MultiTrace 1.0

The information in this guide also applies to later versions of these tools.

A suitable target will also be needed, such as an Integrator core module with an ETM connected to the PC via a Multi-ICE unit.

1.1 - Setting up the hardware

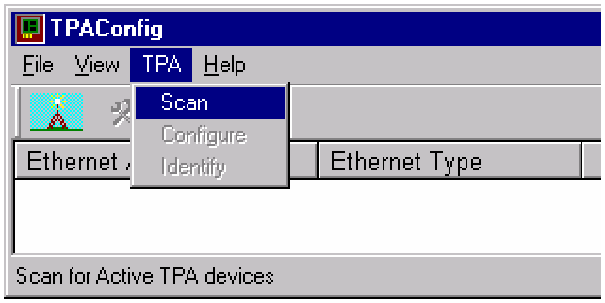
Using the cables provided, connect the MultiTrace unit to the target board, the network and the power supply as follows:



The ‘T piece’ adapter must be used in order to attach the MultiTrace ribbon cable to the MICTOR connector on the target board.

1.2 - Configuring the Ethernet connection

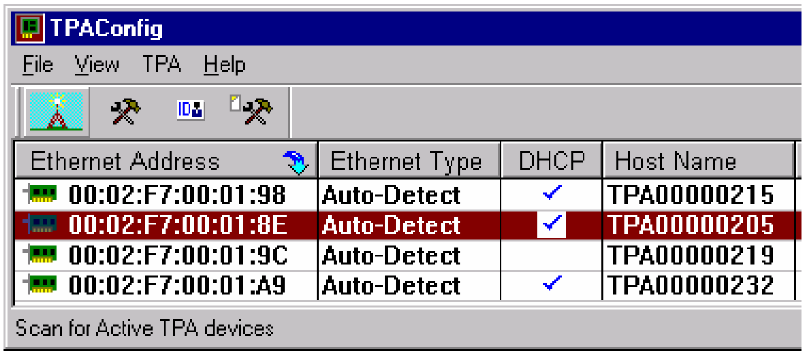
From the *Start* menu select *Programs* → *ARM MultiTrace 1.0* → *Ethernet Config Utility.* You will see a window as shown below:



Scan for Active TPA devices

Scan the local network for MultiTrace units by selecting *TPA* → *Scan* from the menu, or clicking the *Scan* button on the toolbar.

After a few seconds the utility will display the MultiTrace units it has located:



Ethernet Address	Ethernet Type	DHCP	Host Name
00:02:F7:00:01:98	Auto-Detect	✓	TPA00000215
00:02:F7:00:01:8E	Auto-Detect	✓	TPA00000205
00:02:F7:00:01:9C	Auto-Detect		TPA00000219
00:02:F7:00:01:A9	Auto-Detect	✓	TPA00000232

Scan for Active TPA devices

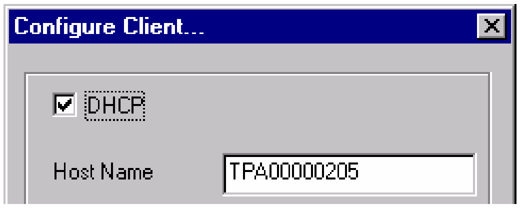
If there are several MultiTrace units listed the correct unit can be identified by the *Ethernet Address* and the *Host Name* numbers, both of which correspond to numbers that are printed on a label on the underside of the MultiTrace unit.

A MultiTrace unit with a *Host Name* of **`TPA00000205`** will have **`S/N:205`** printed on the label on the base of the unit.

Confirm the correct MultiTrace unit is chosen by highlighting it then selecting *TPA* → *Identify* from the menu or clicking on the *ID* button on the toolbar.

The selected MultiTrace unit will flash all three LEDs for a few seconds.

Check the configuration settings of the MultiTrace unit by selecting *TPA* → *Configure* from the menu or clicking on the *Configure* button on the toolbar. The *Configure Client…* dialog is displayed (see over).



Configure Client...

☑ DHCP

Host Name TPA00000205

Ensure the *DHCP* checkbox is ticked, then click on *Configure* to close the dialog.

The MultiTrace unit is now ready for use with the debug tools. Close the TPAConfig utility before proceeding to the next section.

Section 2: Using MultiTrace with ADS

This section provides an introduction to using the MultiTrace unit with ADS to perform trace capture.

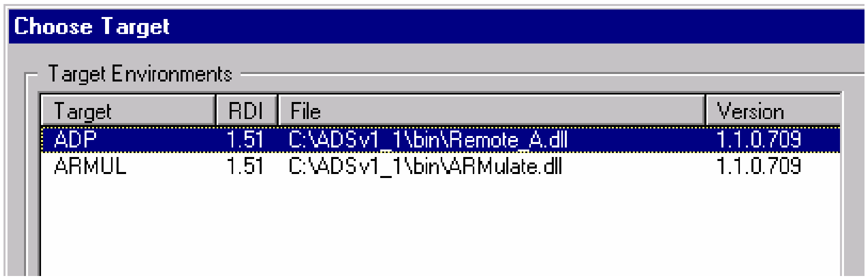
For full, detailed information on this topic please refer to the MultiTrace user guide.

2.1 - Configuring the Target

Ensure the Multi-ICE server is running and connected to the target board.

Launch AXD by selecting *Programs* → *ARM Developer Suite v1.2* → *AXD Debugger* from the Windows *Start* menu.

Select *Options* → *Configure Target* from the menu. The *Choose Target* dialog appears:

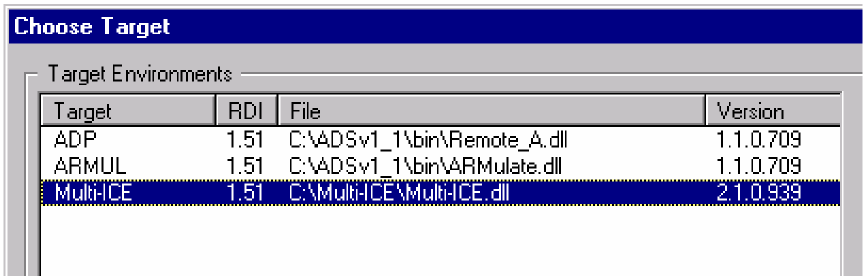


Choose Target

Target Environments

Target	RDI	File	Version
ADP	1.51	C:\ADSv1_1\bin\Remote_A.dll	1.1.0.709
ARMUL	1.51	C:\ADSv1_1\bin\ARMulate.dll	1.1.0.709

Select the *Multi-ICE.dll* if available, if not click *Add* to locate the *Multi-ICE.dll* (found in the Multi-ICE install directory) and then select it:



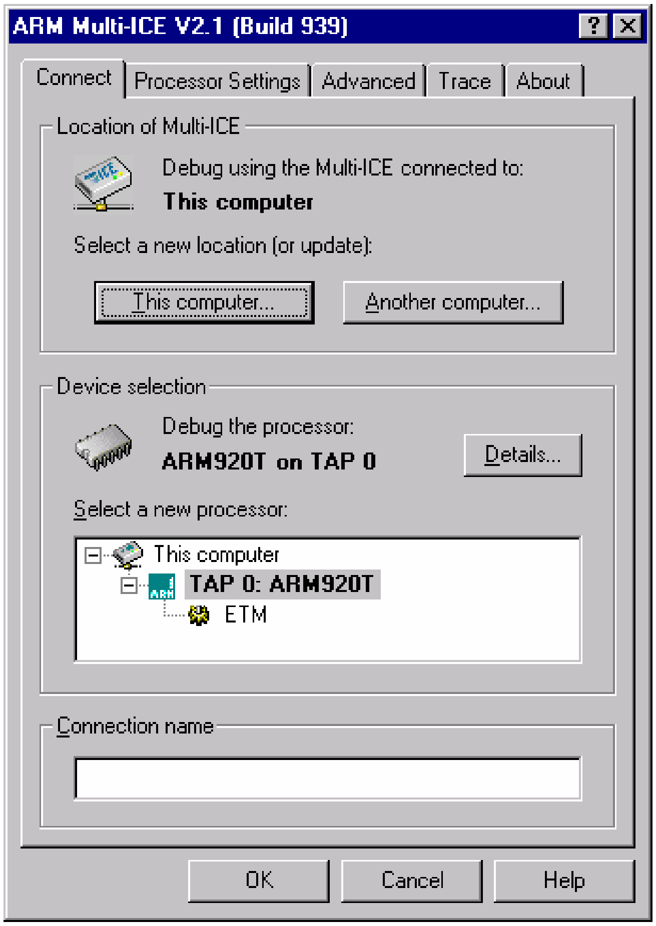
Choose Target

Target Environments

Target	RDI	File	Version
ADP	1.51	C:\ADSv1_1\bin\Remote_A.dll	1.1.0.709
ARMUL	1.51	C:\ADSv1_1\bin\ARMulate.dll	1.1.0.709
Multi-ICE	1.51	C:\Multi-ICE\Multi-ICE.dll	2.1.0.939

Click *Configure*, to open the Multi-ICE configuration window (see below)

If you see a *Multi-ICE Welcome* dialog, click *OK*.



Location of Multi-ICE

Debug using the Multi-ICE connected to:
This computer

Select a new location (or update):

Device selection

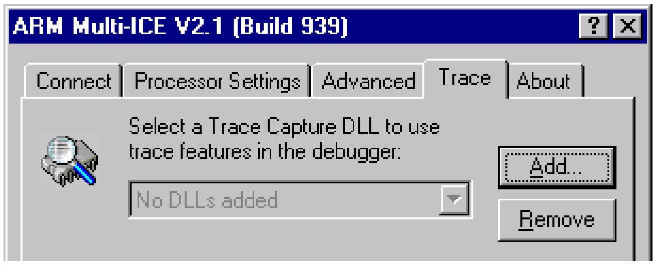
Debug the processor:
ARM920T on TAP 0

Select a new processor:

- This computer
  - TAP 0: ARM920T
    - ETM

Connection name

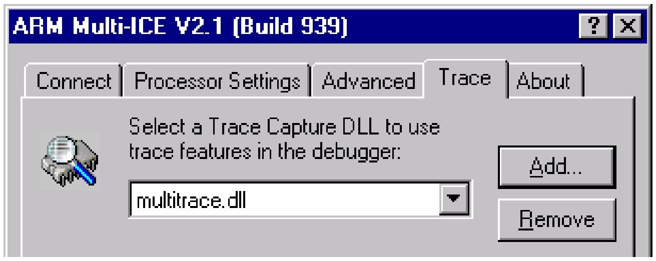
Ensure *This computer* is selected as the location for the Multi-ICE connection and that the *Device selection* frame shows the correct target, then click on the *Trace* tab:



Select a Trace Capture DLL to use trace features in the debugger:

No DLLs added

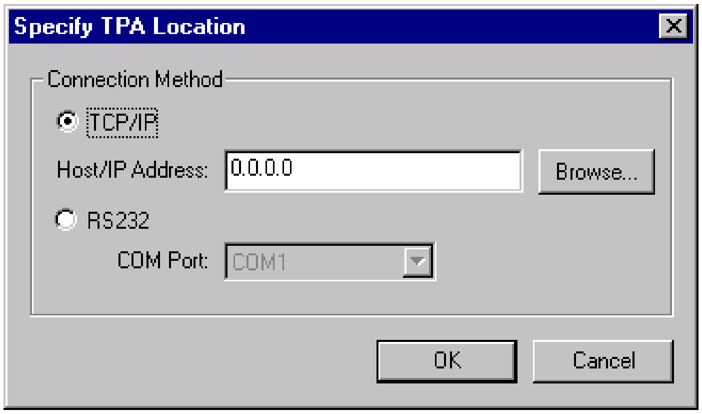
Click *Add* to locate the *MultiTrace.dll* (found in the MultiTrace install directory):


ARM Multi-ICE V2.1 (Build 939)

Select a Trace Capture DLL to use trace features in the debugger:

multitrace.dll

Click *Change Location* to open the *Specify TPA Location* dialog:


Specify TPA Location

Connection Method

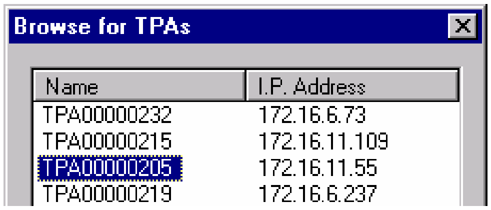
● TCP/IP

Host/IP Address: 0.0.0.0

○ RS232

COM Port: COM1

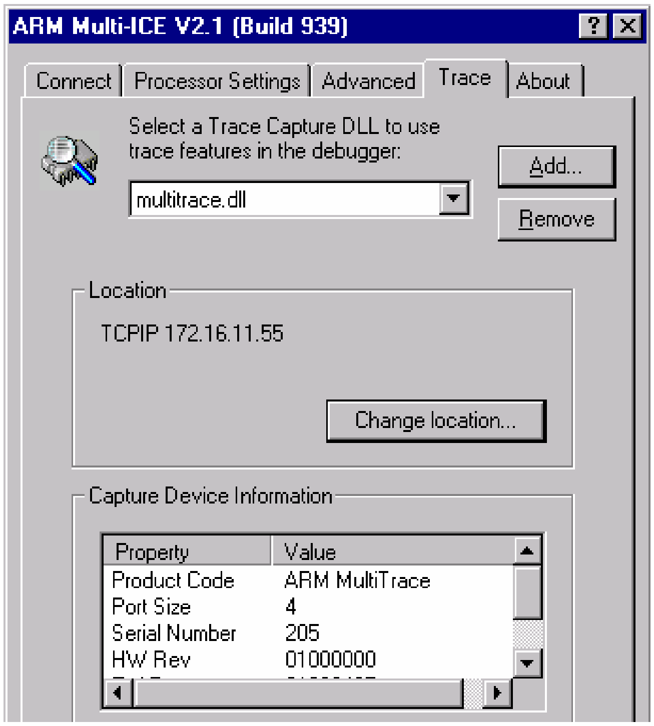
Click *Browse* to open the *Browse* dialog:


Browse for TPAs

Name	I.P. Address
TPA00000232	172.16.6.73
TPA00000215	172.16.11.109
TPA00000205	172.16.11.55
TPA00000219	172.16.6.237

Select the appropriate MultiTrace unit from the list and click *OK*.

Click *OK* again to close the *Specify TPA Location* dialog and return to the Multi-ICE configuration window:



Select a Trace Capture DLL to use trace features in the debugger:

multitrace.dll

Location

TCPIP 172.16.11.55

Capture Device Information

Property	Value
Product Code	ARM MultiTrace
Port Size	4
Serial Number	205
HW Rev	01000000

Click *OK*, then *OK* again to close both the Multi-ICE configuration window and the *Choose Target* dialog.

2.2 – Performing Trace Capture

This section uses two simple examples to provide an introduction to using the tools to perform trace capture.

The files needed for these examples can be found in:
`c:\ads_tutorial\trace\`

Example 1 – Tracing a function

This is a simple example that shows how trace capture can be done for a specific function, `writefunc`.

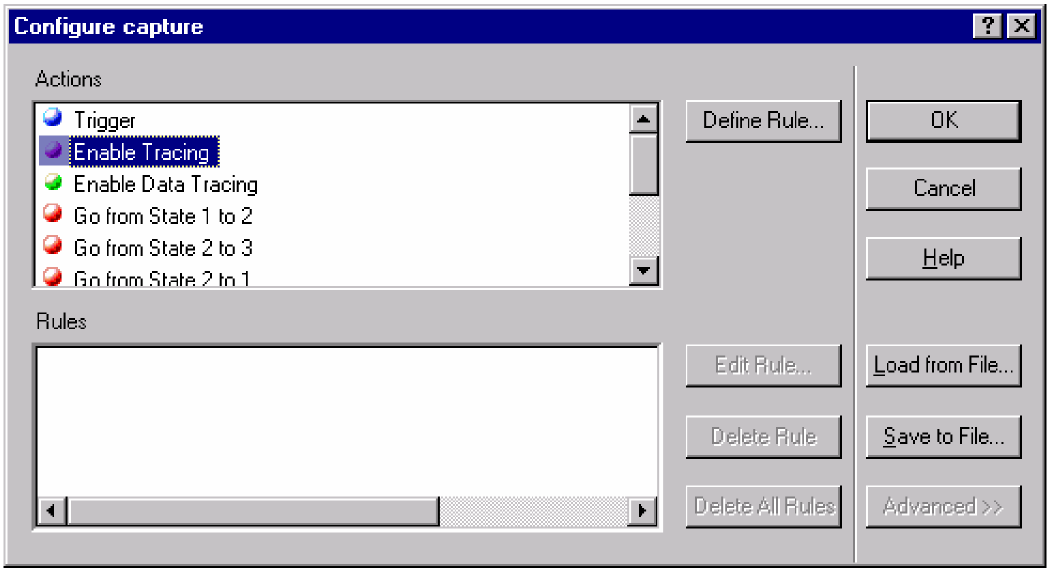
Select *System Views* → *Debugger Internals* from the menu to open the *Debugger Internals* window.

Double click in the *Value* column of the `$top_of_memory` variable and enter `0x40000` as the new value.

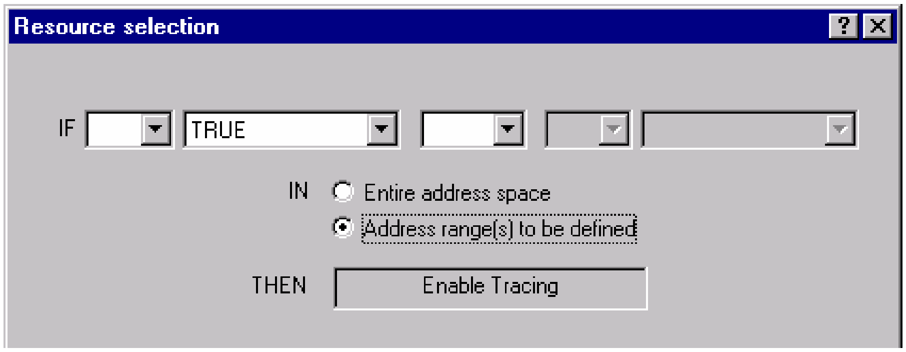
Load the example image `func.axf` by selecting *File* → *Load Image* from the menu or clicking on the *Load Image* button on the toolbar.

Open the *Trace* view window by selecting *Processor Views* → *Trace* → *View Trace* from the menu.

Open the *Configure capture* dialog by selecting *Processor Views* → *Trace* → *Configure Capture* from the menu:



Highlight *Enable Tracing* in the actions list, then click the *Define Rule…*button to display the *Resource selection* dialog box.

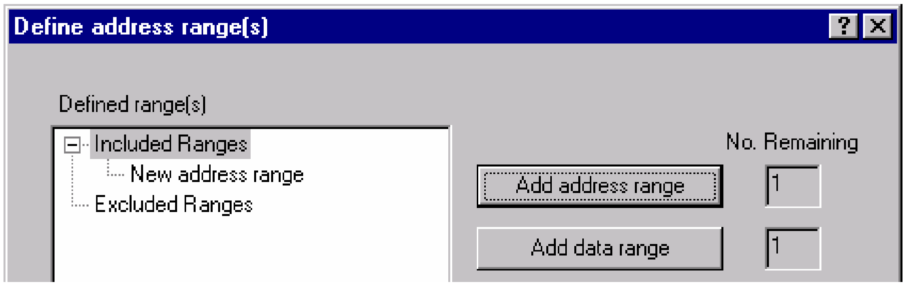


IF TRUE

IN ○ Entire address space
● Address range(s) to be defined

THEN Enable Tracing

Select the *Address Ranges to be Defined* radio button then click *Next*.

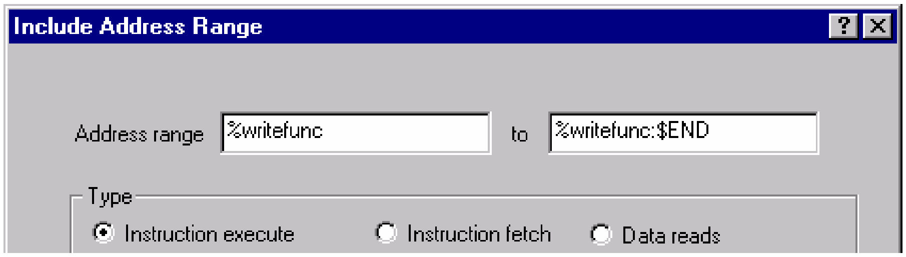


Defined range(s)

⊟ Included Ranges
New address range
Excluded Ranges

No. Remaining: 1 (Add address range), 1 (Add data range)

Ensure *Included Ranges* is highlighted and click *Add address range*, then click *Next*.



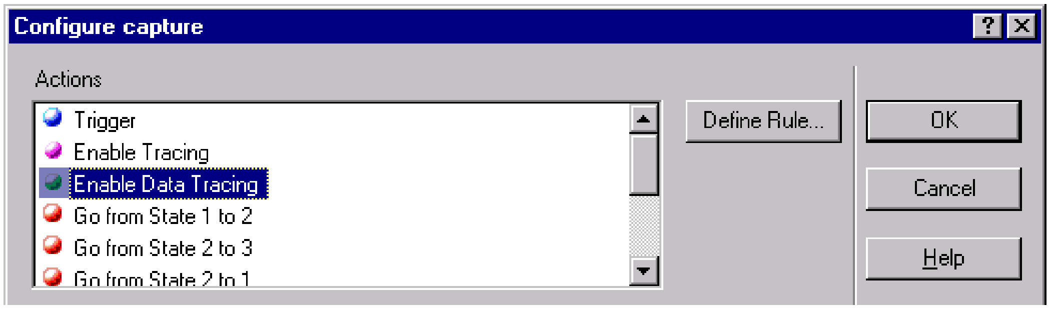
Address range %writefunc to %writefunc:$END

Type
● Instruction execute ○ Instruction fetch ○ Data reads

Ensure the *Instruction execute* radio button is selected and make the following entries:

Address range: **`%writefunc`**
to: **`%writefunc:$END`**

Click *Next*, then click *Finish* to return to the *Configure capture* dialog:

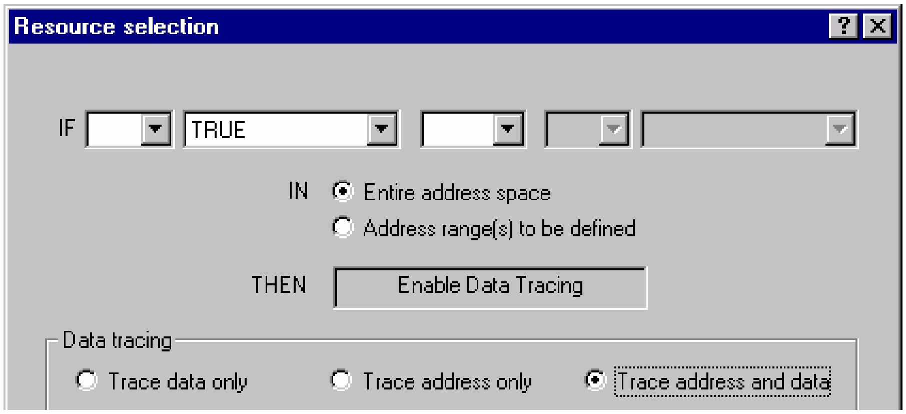


Configure capture

Actions

- Trigger
- Enable Tracing
- Enable Data Tracing
- Go from State 1 to 2
- Go from State 2 to 3
- Go from State 2 to 1

Highlight *Enable Data Tracing* and click on *Define Rule…*



Resource selection

IF | TRUE

IN ● Entire address space
○ Address range(s) to be defined

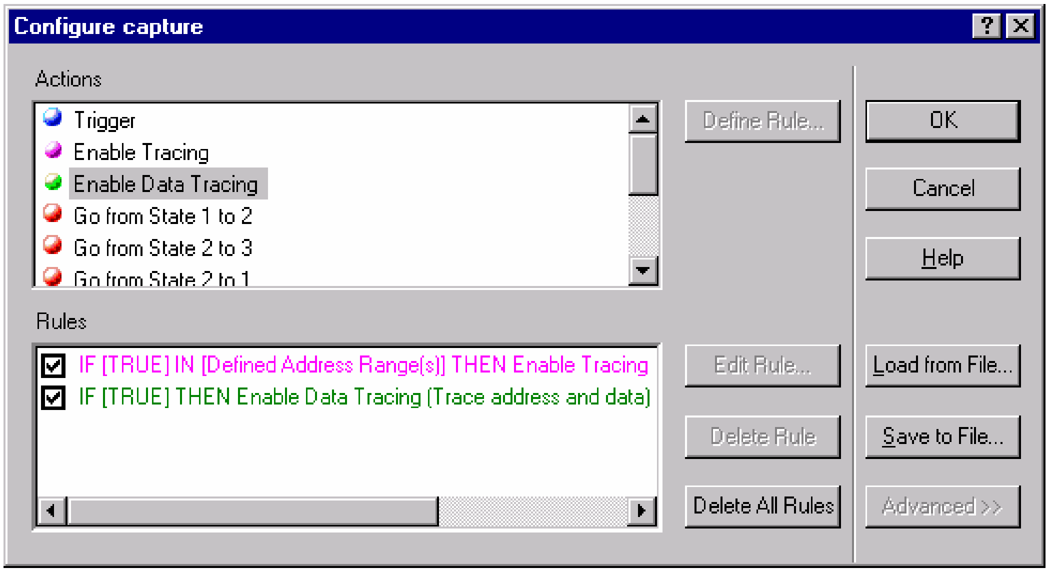
THEN Enable Data Tracing

Data tracing

○ Trace data only ○ Trace address only ● Trace address and data

Ensure the *Entire address space* and *Trace address and data* radio buttons are selected, then click *Next*.

Click *Finish* to return to the *Configure capture dialog*:

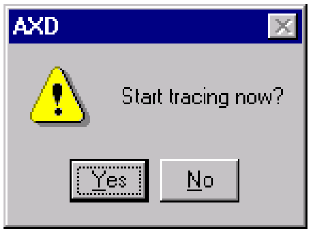


- Trigger
- Enable Tracing
- Enable Data Tracing
- Go from State 1 to 2
- Go from State 2 to 3
- Go from State 2 to 1

☑ IF [TRUE] IN [Defined Address Range(s)] THEN Enable Tracing

☑ IF [TRUE] THEN Enable Data Tracing (Trace address and data)

The necessary trace capture rules are now completely defined. Click *OK* to close the *Configure capture dialog*.



Start tracing now?

Click *Yes* when asked if you would like to start tracing.

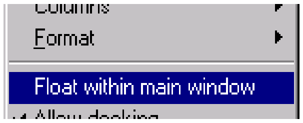
Start executing the image by selecting *Execute* → *Go* (*F5*) from the AXD menu.

Execution halts at the breakpoint on main. Select *Go* again to continue execution.

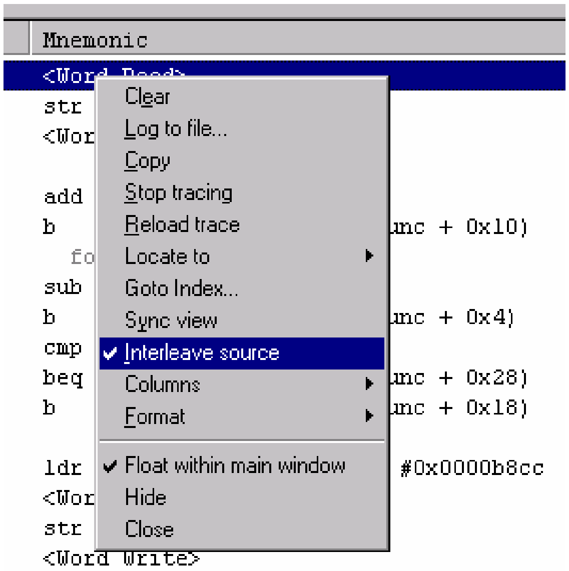
Enter a number between `1` and `10` when prompted for input.

Execution halts and the trace tools load the trace window with data reconstructed from the trace capture buffer.

To view the trace data more clearly right click inside the trace window pane and select *Float within main window* from the context menu.



View the C source code interleaved with the traced instructions by right clicking in the trace window and selecting *Interleave source* from the context menu.



Note how the instructions traced correspond to the operation of **`writefunc`**. The successive writes to memory of the incremental values can also be clearly seen as the **`<Word Write>`** entries.

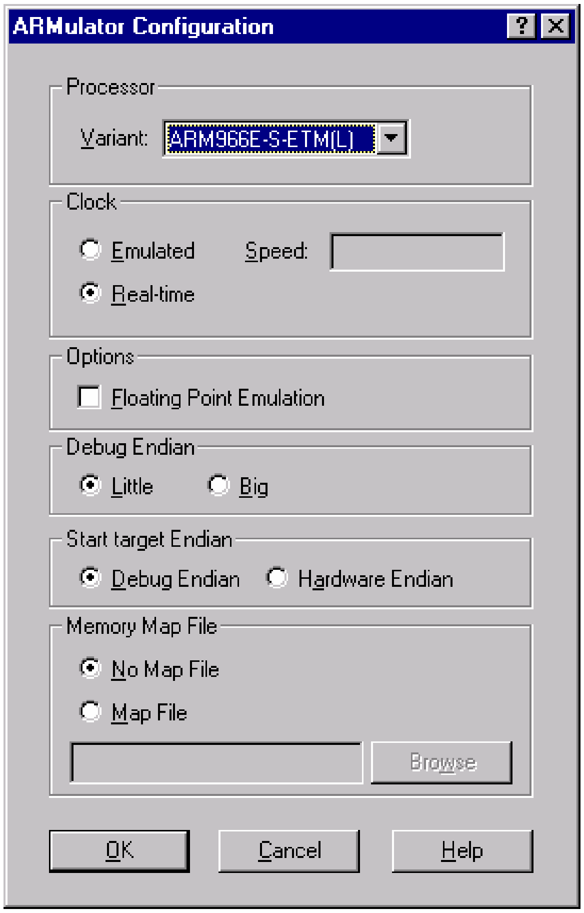
Example 2 – Tracing a write to a global variable

This example shows how the ARMulator can be used to capture trace and how trace capture can be turned on and off according to the value of a global variable.

Select *Options* → *Configure Target* from the AXD menu.

Select *ARMUL* as the target and click *Configure*.

From the drop-down box select *ARM966E-S-ETM(L)* as your target processor:



ARMulator Configuration

Processor

Variant: ARM966E-S-ETM(L)

Clock

○ Emulated Speed:

● Real-time

Options

□ Floating Point Emulation

Debug Endian

● Little ○ Big

Start target Endian

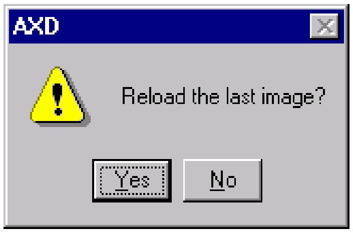
● Debug Endian ○ Hardware Endian

Memory Map File

● No Map File

○ Map File

Click *OK*, then *OK* again



Reload the last image?

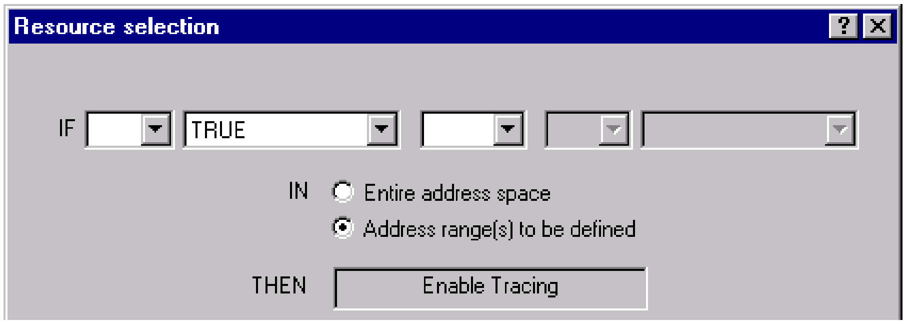
Click *No* when asked if you would like to reload the image.

Load the example image **`glob.axf`** by selecting *File* → *Load Image* from the menu or clicking on the *Load Image* button on the toolbar.

Open the *Configure capture* dialog by selecting *Processor Views* → *Trace* → *Configure Capture* from the menu:

Remove the current trace capture rules by clicking on *Delete All Rules*.

Highlight *Enable Tracing* and click *Define Rule*



IF TRUE

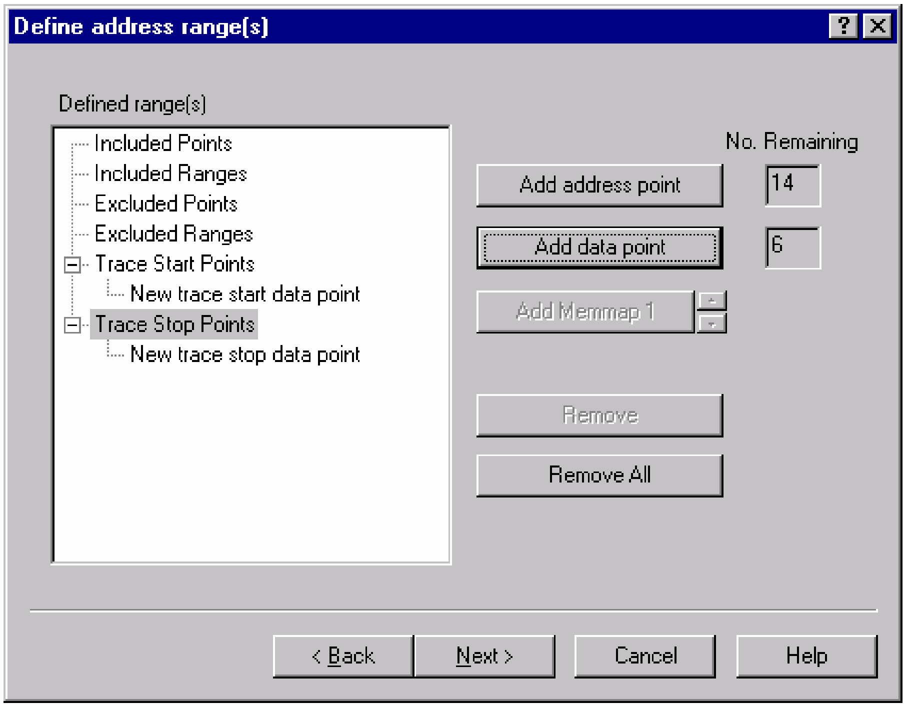
IN ○ Entire address space
● Address range(s) to be defined

THEN Enable Tracing

In the *Resource Selection* dialog select the *Address Range(s) to be Defined* radio button then click *Next*.

In the *Define address range(s)* dialog highlight *Trace Start Points* and click on *Add data point*

In the *Define address range(s)* dialog highlight *Trace Stop Points* and click on *Add data point*

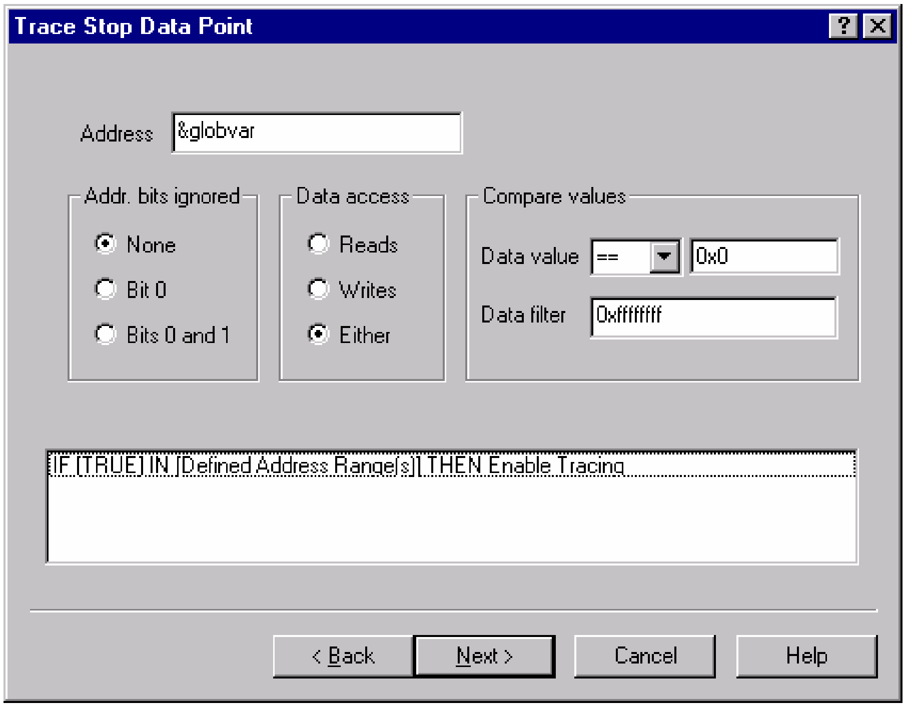


Click *Next* and make the following entries in the *Trace Start Data Point* dialog:

Address:	`&globvar`
Addr. bits ignored:	`None`
Data access:	`Writes`
Data value:	`== 0x1`
Data filter:	`0xffffffff`

Click *Next* and use the same details to define the *Trace Stop Data Point* except this time use the compare value:

Data value: `==0x0`



Trace Stop Data Point

Address: &globvar

Addr. bits ignored: ● None ○ Bit 0 ○ Bits 0 and 1

Data access: ○ Reads ○ Writes ● Either

Compare values:

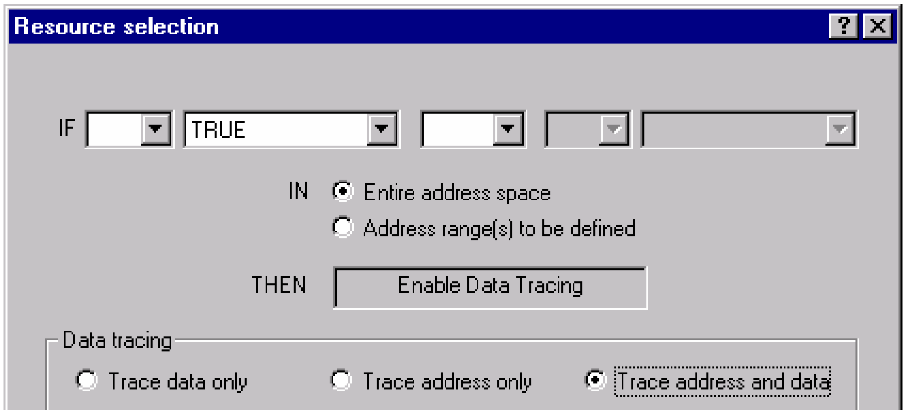
Data value: == 0x0

Data filter: 0xffffffff

IF [TRUE] IN [Defined Address Range(s)] THEN Enable Tracing

Click *Next* to completely define the rule, then click *Finish* to return to the *Configure capture* dialog.

Highlight *Enable Data Tracing* and click on *Define Rule…*



IF TRUE

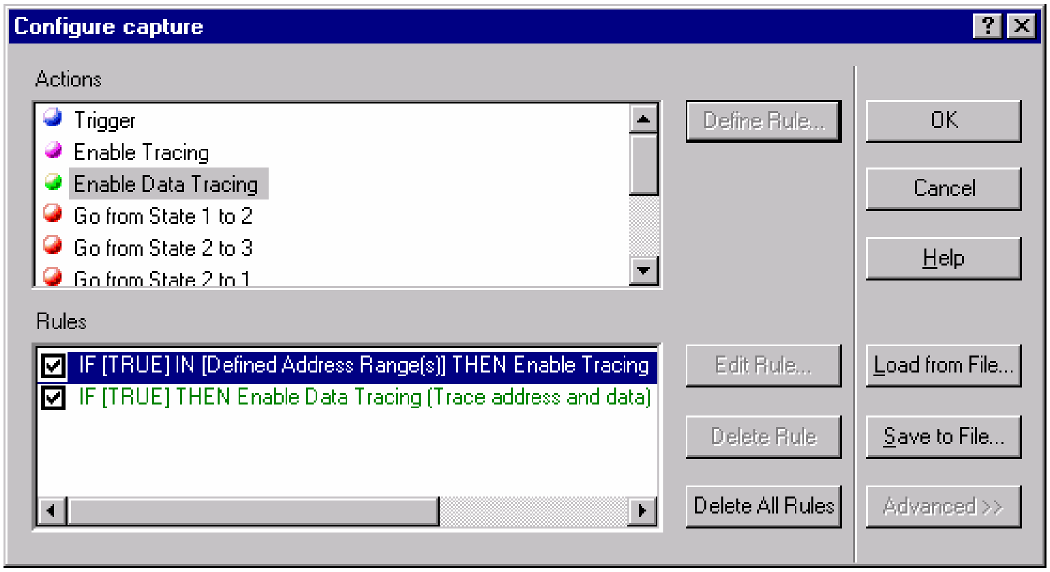
IN ● Entire address space ○ Address range(s) to be defined

THEN Enable Data Tracing

Data tracing: ○ Trace data only ○ Trace address only ● Trace address and data

Ensure the *Entire address space* and *Trace address and data* radio buttons are checked, then click *Next*.

Click *Finish* to return to the *Configure capture dialog*:



Actions:

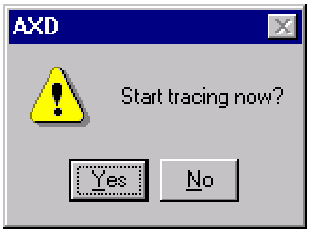
- Trigger
- Enable Tracing
- Enable Data Tracing
- Go from State 1 to 2
- Go from State 2 to 3
- Go from State 2 to 1

Rules:

☑ IF [TRUE] IN [Defined Address Range(s)] THEN Enable Tracing

☑ IF [TRUE] THEN Enable Data Tracing (Trace address and data)

The necessary trace capture rules are now completely defined. Click *OK* to close the *Configure capture dialog*.



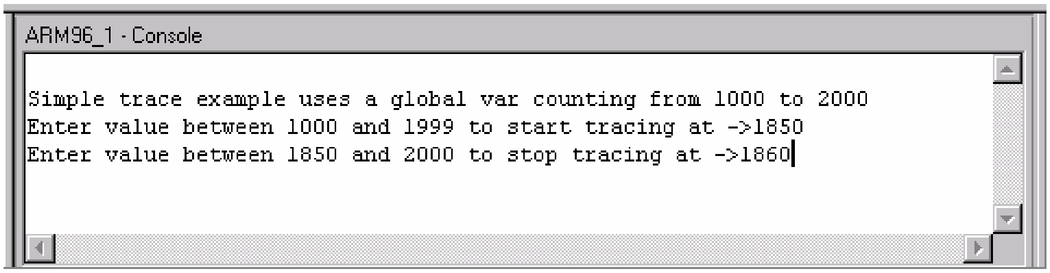
Start tracing now?

Click *Yes* when asked if you would like to start tracing.

Start executing the image by selecting *Execute* → *Go* (*F5*) from the AXD menu.

Execution halts at the breakpoint on main. Select *Go* again to continue execution.

When prompted enter two values at which to start and then stop trace capture:

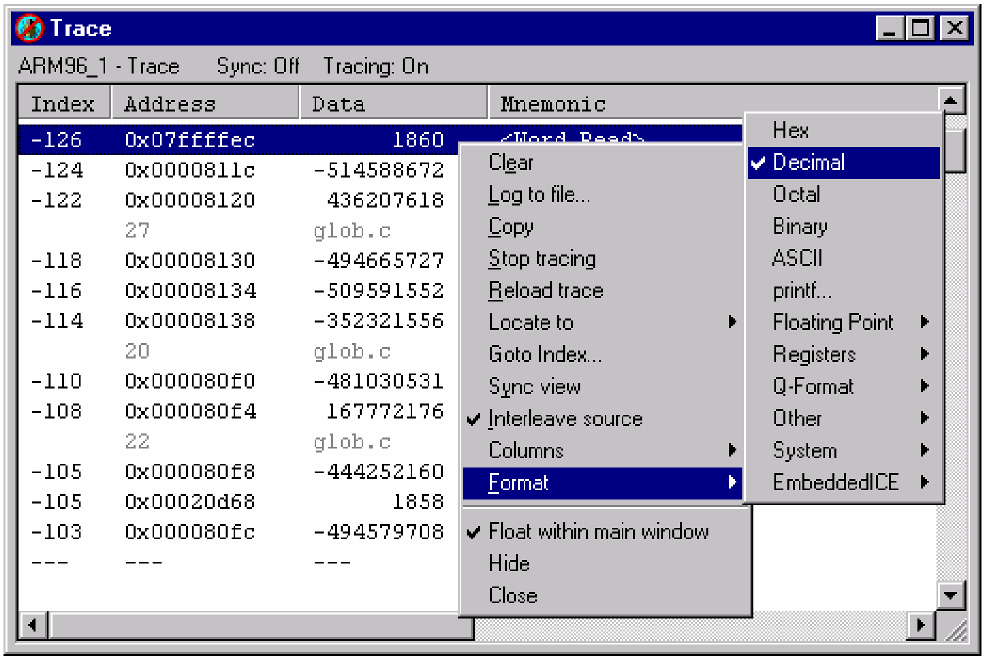


```
Simple trace example uses a global var counting from 1000 to 2000
Enter value between 1000 and 1999 to start tracing at ->1850
Enter value between 1850 and 2000 to stop tracing at ->1860
```

View the trace capture in the trace window

Right click inside the trace window and ensure the *Interleave Source* option is checked

Right click in the *Data* column and select *Format*→*Decimal*



Trace

ARM96_1 - Trace   Sync: Off   Tracing: On

Index	Address	Data	Mnemonic
-126	0x07ffffec	1860	<Word Read>
-124	0x0000811c	-514588672	
-122	0x00008120	436207618	
	27	glob.c	
-118	0x00008130	-494665727	
-116	0x00008134	-509591552	
-114	0x00008138	-352321556	
	20	glob.c	
-110	0x000080f0	-481030531	
-108	0x000080f4	167772176	
	22	glob.c	
-105	0x000080f8	-444252160	
-105	0x00020d68	1858	
-103	0x000080fc	-494579708	
---	---	---	

Note the instructions and data **`<Word Writes>`** traced correspond to the values chosen.

☒ Run the example again with different input values and trace capture rules, or close AXD to finish the tutorial.