(DEBUGGING) Setup and Run the Debugger

Make Target
In order to use the debugger, create a new make target for an existing project so that *–debug* is added after the platform name (with no space) and remove run from the end of the target. That is, the target should look like:

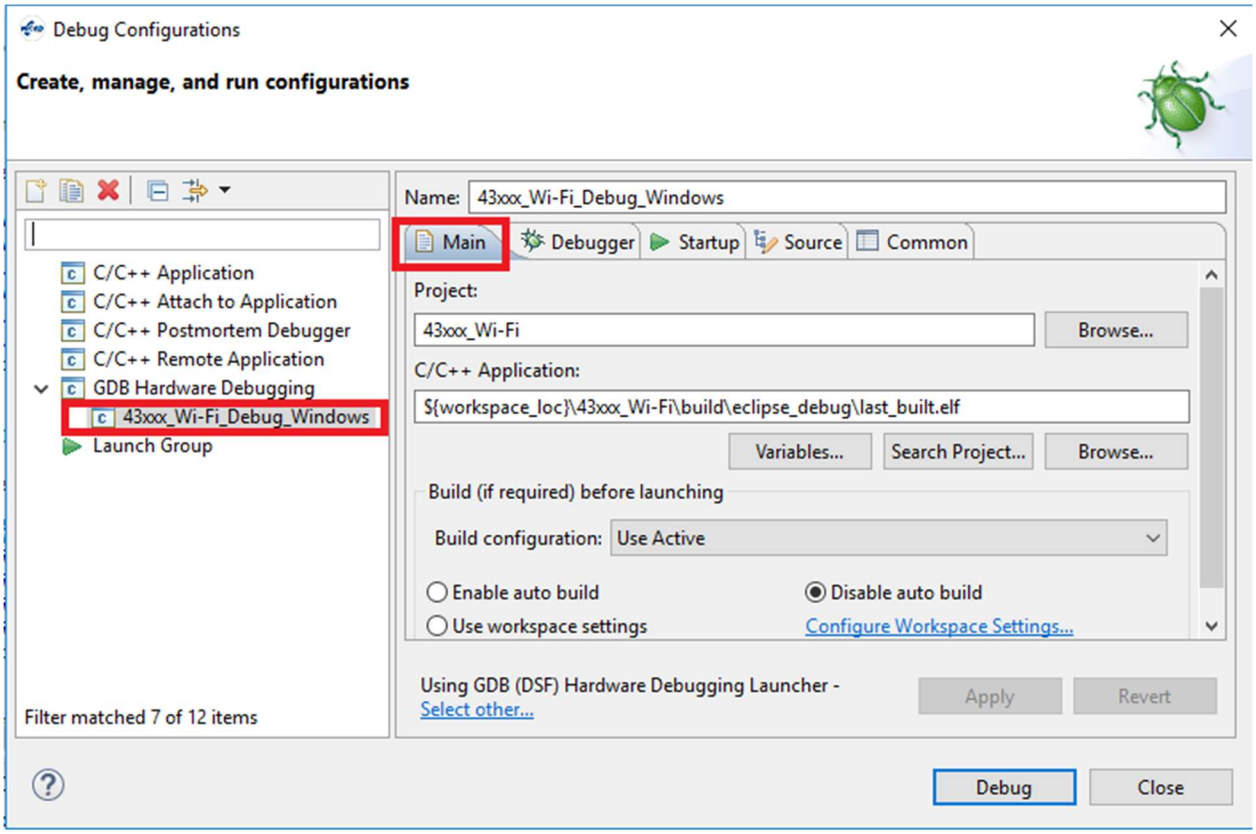
<folder1>.[<folder2>…].<project>-<platform>-debug download

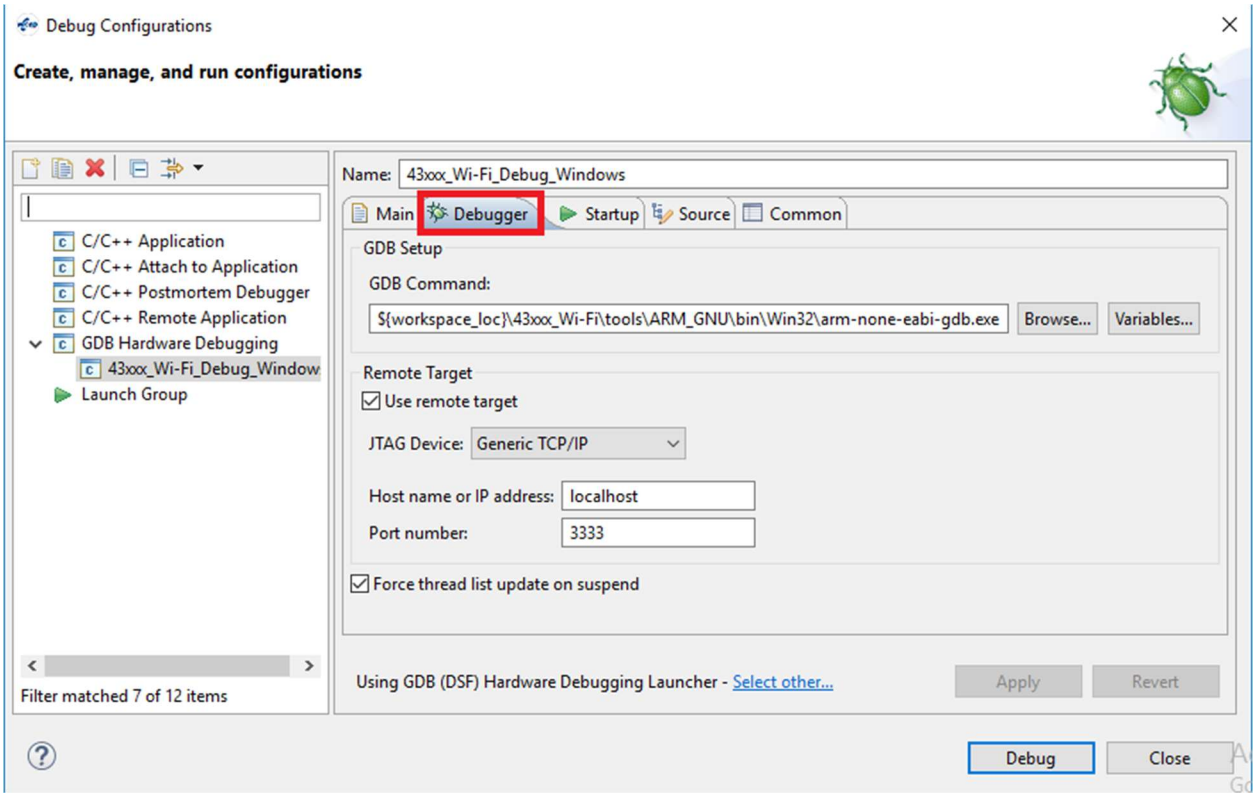
For example, the make target for the **scan** project from the folder snip would be:

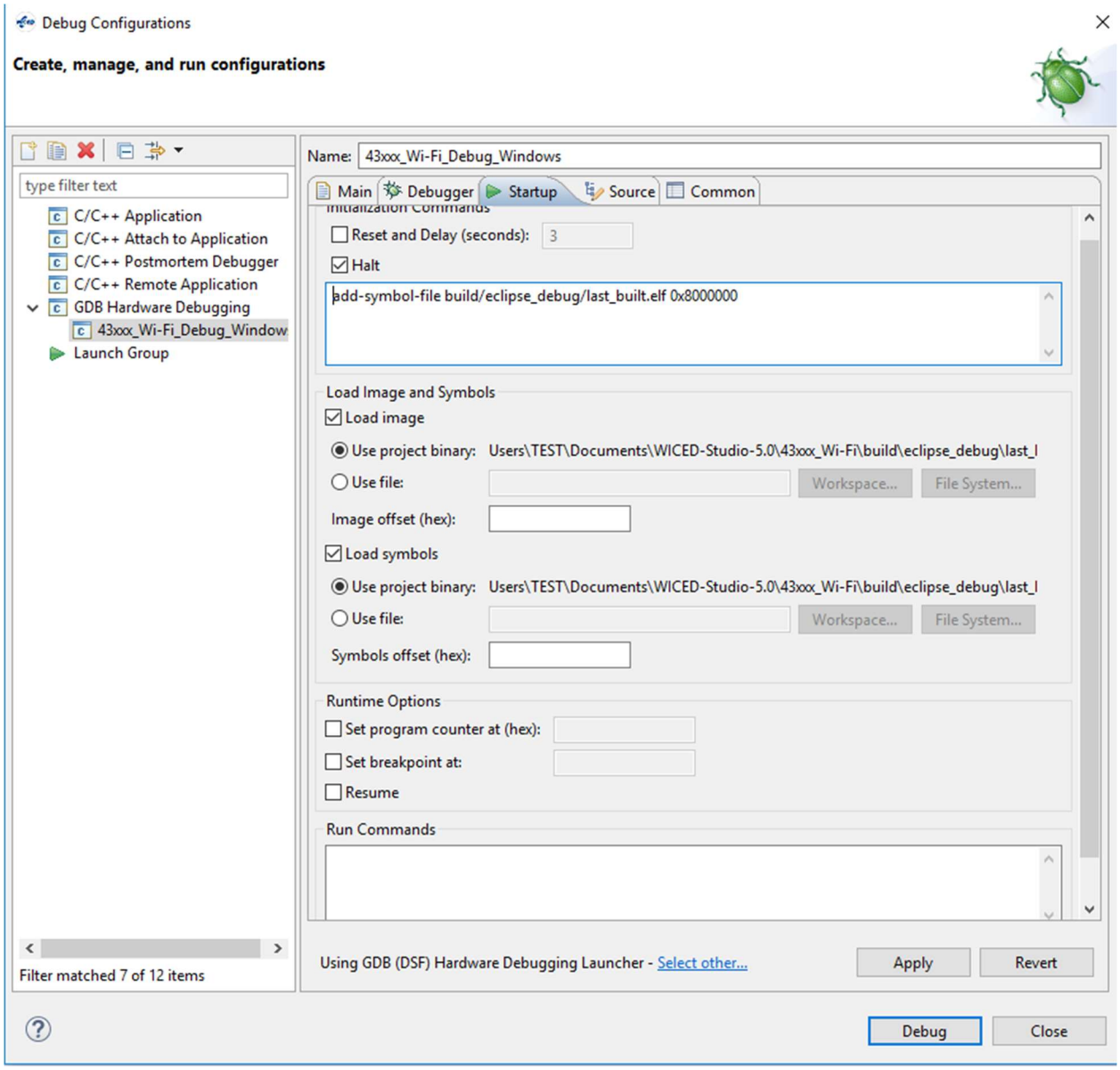
snip.scan-CYW943907AEVAL1F -debug download

Setup

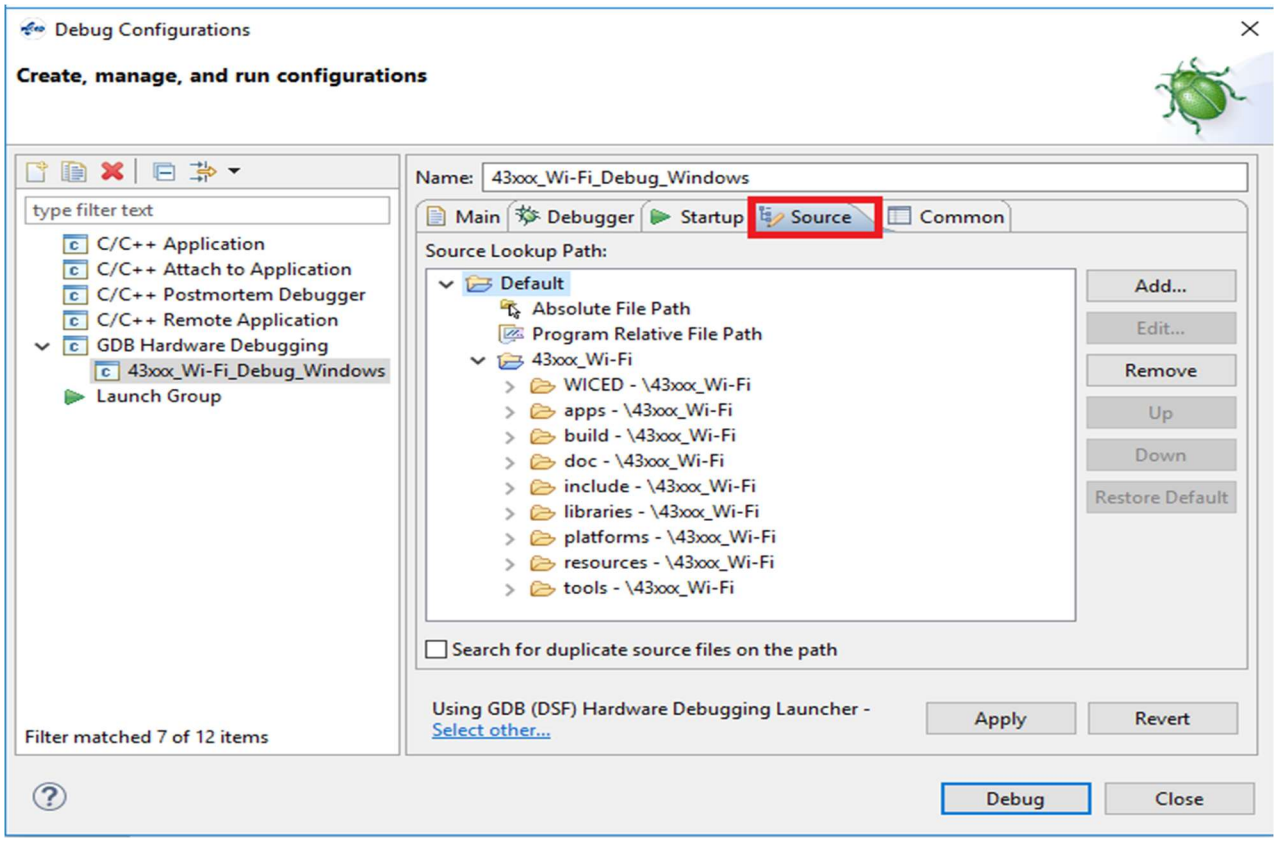
Before starting the debugger, we need to verify that it is setup correctly. From WICED Studio, click the down arrow next to the green bug icon and select “Debug Configurations…” Then select “GDB Hardware Debugging > 43xxx_Wi-Fi Debug_Windows” from the window on the left. Setup the various tabs as shown in the figures below. You should only have to make changes on the “Startup” and “Common” tabs but all are shown here for completeness.

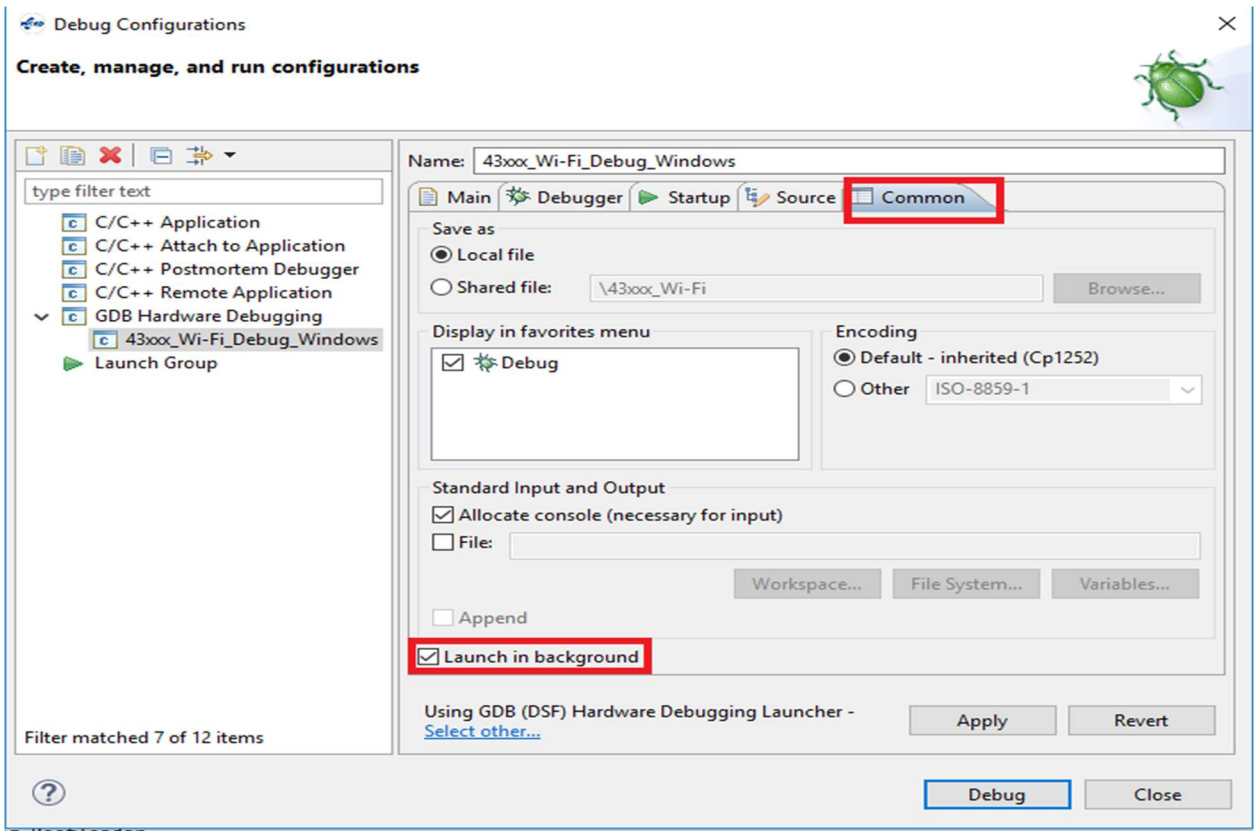



Debug Configurations
Create, manage, and run configurations
C/C++ Application
C/C++ Attach to Application
C/C++ Postmortem Debugger
C/C++ Remote Application
GDB Hardware Debugging
43xxx_Wi-Fi_Debug_Window
Launch Group
Filter matched 7 of 12 items
Name: 43xxx_Wi-Fi_Debug_Windows
Main
Debugger
Startup
Source
Common
GDB Setup
GDB Command:
${workspace_loc}\43xxx_Wi-Fi\tools\ARM_GNU\bin\Win32\arm-none-eabi-gdb.exe
Browse...
Variables...
Remote Target
Use remote target
JTAG Device:
Generic TCP/IP
Host name or IP address:
localhost
Port number:
3333
Force thread list update on suspend
Using GDB (DSF) Hardware Debugging Launcher - Select other...
Apply
Revert
Debug
Close



Note: the text in the box above is: ***add-symbol-file build/eclipse_debug/last_built.elf 0x8000000***

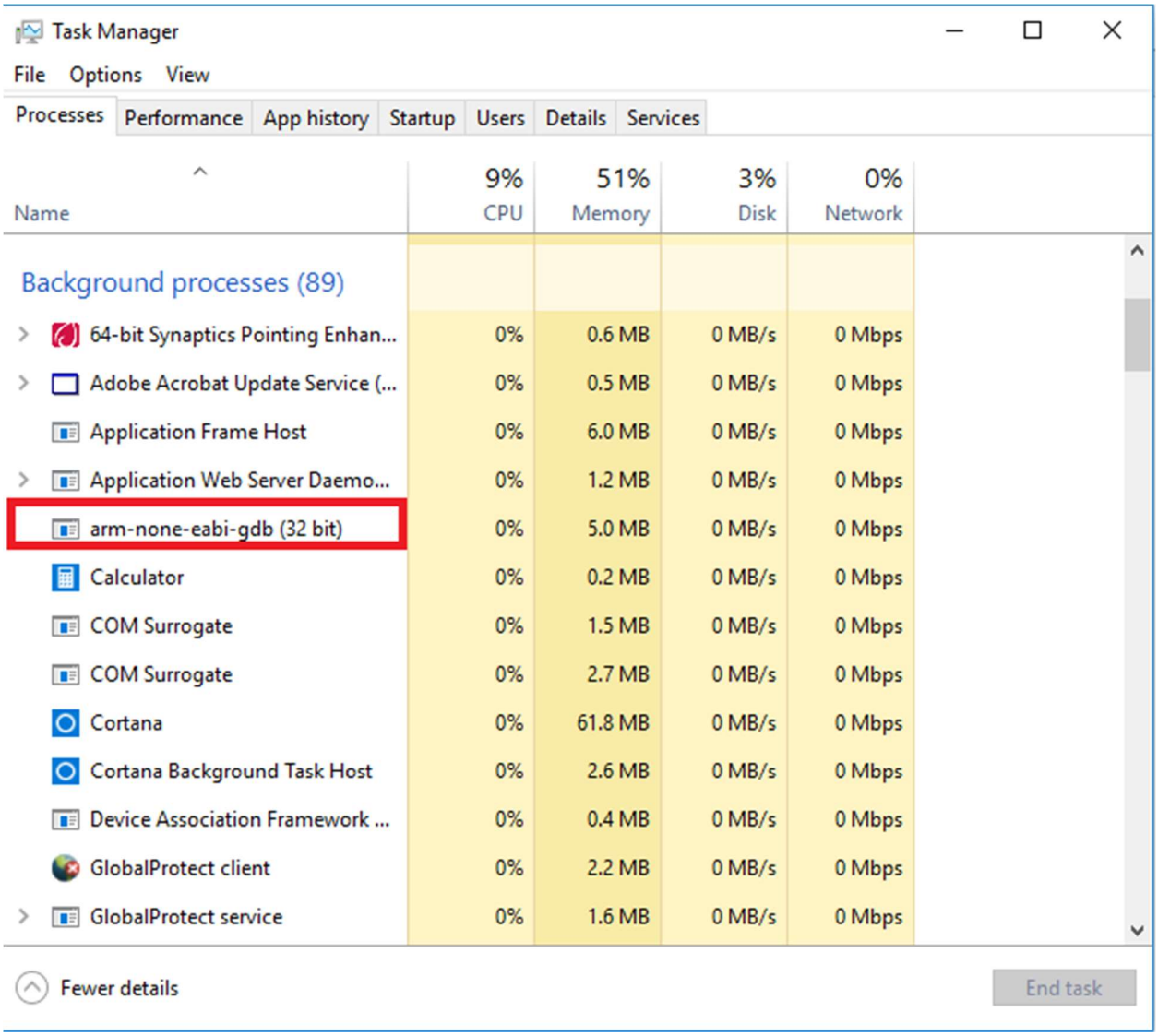

Debug Configurations
Create, manage, and run configurations
type filter text
C/C++ Application
C/C++ Attach to Application
C/C++ Postmortem Debugger
C/C++ Remote Application
GDB Hardware Debugging
43xxx_Wi-Fi_Debug_Windows
Launch Group
Filter matched 7 of 12 items
Name: 43xxx_Wi-Fi_Debug_Windows
Main
Debugger
Startup
Source
Common
Source Lookup Path:
Default
Absolute File Path
Program Relative File Path
43xxx_Wi-Fi
WICED - \43xxx_Wi-Fi
apps - \43xxx_Wi-Fi
build - \43xxx_Wi-Fi
doc - \43xxx_Wi-Fi
include - \43xxx_Wi-Fi
libraries - \43xxx_Wi-Fi
platforms - \43xxx_Wi-Fi
resources - \43xxx_Wi-Fi
tools - \43xxx_Wi-Fi
Add...
Edit...
Remove
Up
Down
Restore Default
Search for duplicate source files on the path
Using GDB (DSF) Hardware Debugging Launcher -
Select other...
Apply
Revert
Debug
Close


Debug Configurations
Create, manage, and run configurations
type filter text
C/C++ Application
C/C++ Attach to Application
C/C++ Postmortem Debugger
C/C++ Remote Application
GDB Hardware Debugging
43xxx_Wi-Fi_Debug_Windows
Launch Group
Filter matched 7 of 12 items
Name: 43xxx_Wi-Fi_Debug_Windows
Main
Debugger
Startup
Source
Common
Save as
Local file
Shared file:
\43xxx_Wi-Fi
Browse...
Display in favorites menu
Debug
Encoding
Default - inherited (Cp1252)
Other
ISO-8859-1
Standard Input and Output
Allocate console (necessary for input)
File:
Workspace...
File System...
Variables...
Append
Launch in background
Using GDB (DSF) Hardware Debugging Launcher -
Select other...
Apply
Revert
Debug
Close

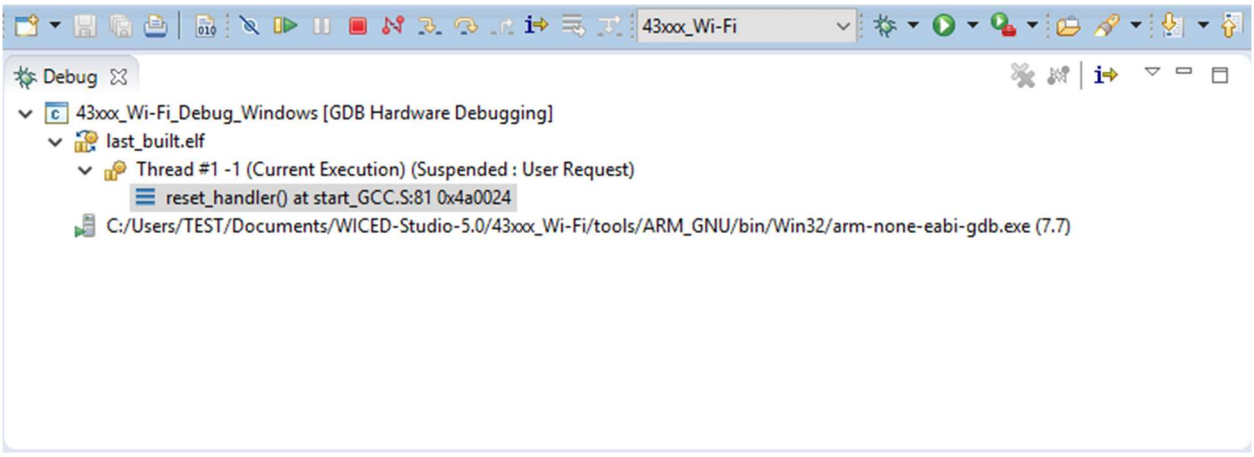
Running the Debugger

Once the setup is complete, execute the make target to download the program to the board. Once the project is downloaded, click the down arrow next to the green bug icon and select “43xxx_Wi-Fi_Debug_Windows”. If you get a message asking if you want to open the debug perspective, click “Yes”. You can click the check box to tell the tool to switch automatically in the future.

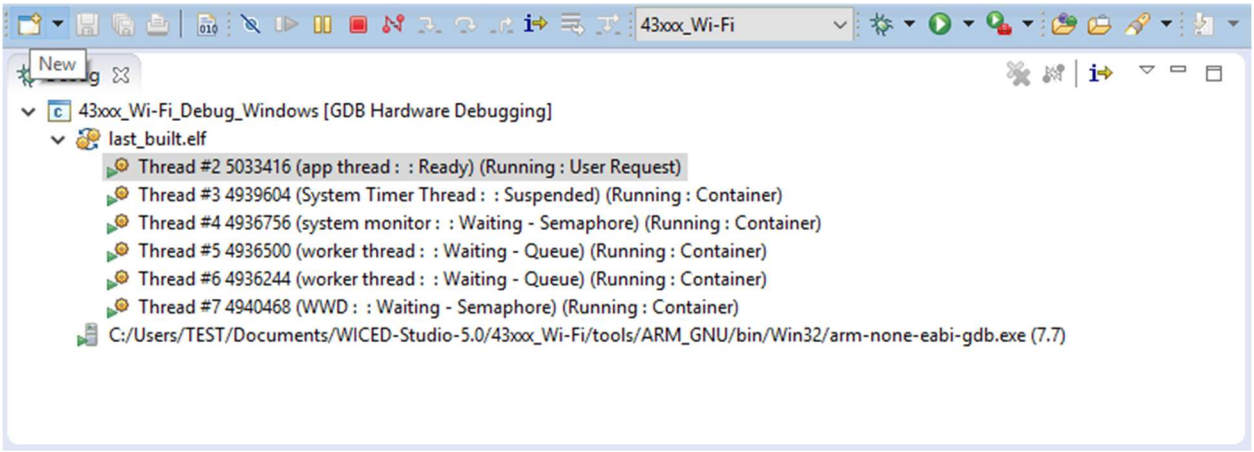
Note: If you get an error when trying to launch the debugger you may need to terminate an existing debug process. Open the Windows Task Manager, select the Process tab, click on “Image Name” to sort by the process name and terminate all “arm-none-eabi-gdb” processes.



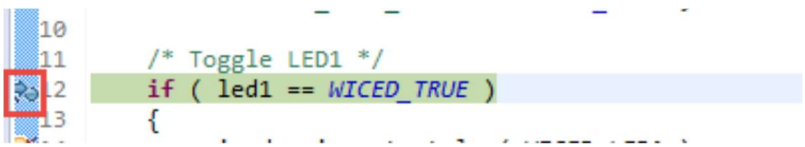
When the debugger starts the top banner will look like this:



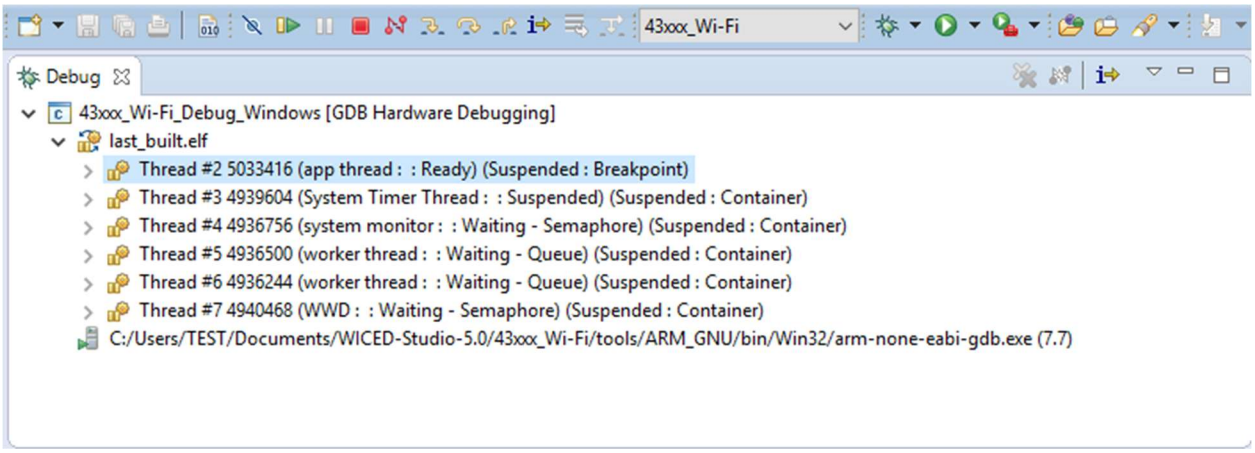
Click the "Resume" (shown in the figure above) until the program continues running and the resume button stays grey. Notice that additional threads along with information about them appears in the debug window.



You can toggle breakpoints by double clicking in the column to the left of the line numbers in the source code or you can right click and select "Toggle Breakpoint". The breakpoint symbol appears to the left of the line number as shown here.



Once a thread suspends due to a breakpoint you will see that line of code highlighted in green as shown above and you will see that the thread is suspended due to the breakpoint in the debug window as shown below.



Click the red “Terminate” button to stop debugging. Once you terminate the debugger, you may want to switch back to the **C/C++** perspective by clicking on the button at the top right corner.