Semihosting in Atollic TrueSTUDIO

Table of content

Before you start…

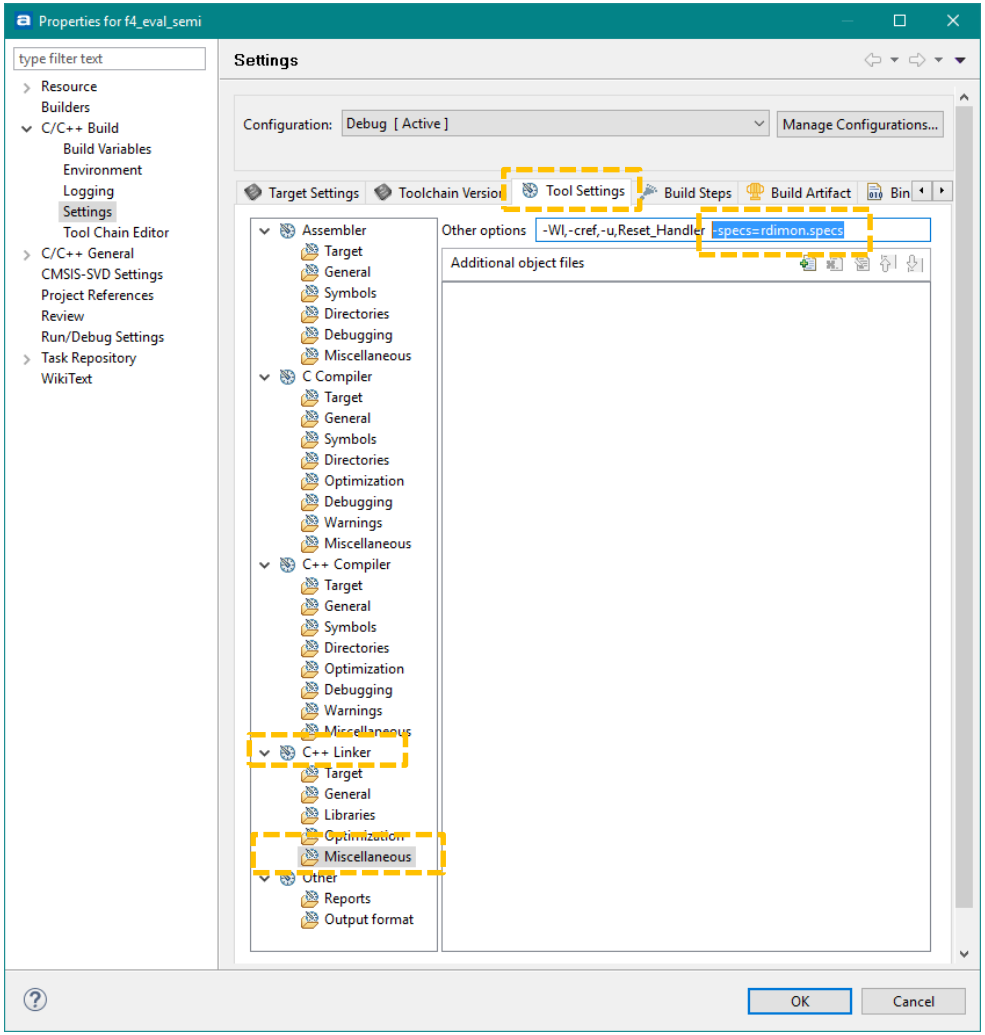
How do I enable semihosting communication in Atollic TrueSTUDIO?

- Semihosting requires some instrumentation in order to output your printf to a PC telnet terminal.
- Semihosting must be enabled in the J-Link GDB-server

What are the limitations using semihosting?

- You need a SEGGER J-Link. Semihosting is not supported by ST-Link GDB-server
- Cortex-M target uses breakpoints (BKPT instruction) to pause target and print messages to terminal. ARM7/ARM9 instead uses software interrupts or supervisor calls (SWI/SVC instructions).
- Application or interrupt code will NOT run while semihosting transfers are active.
- You risk losing interrupts.
- Timing penalties to application execution due to breakpoint/interrupt and due to instrumentation

Linking semihosting instrumented syscalls



1. Open the *project properties* → *tool settings*, go to: *C/C++ Linker* → *Miscellaneous node* and add: ***-specs=rdimon.specs***
Code will now link against semihosting instrumented syscalls

2. Remove any syscalls.c that may be part of the project since such implementation would override the linked syscalls

Initialize the monitor handles

Initialize the semihosting monitor handles before any calls to printf() are being made. In your startup code or in the beginning of main().

For C projects:

main.c

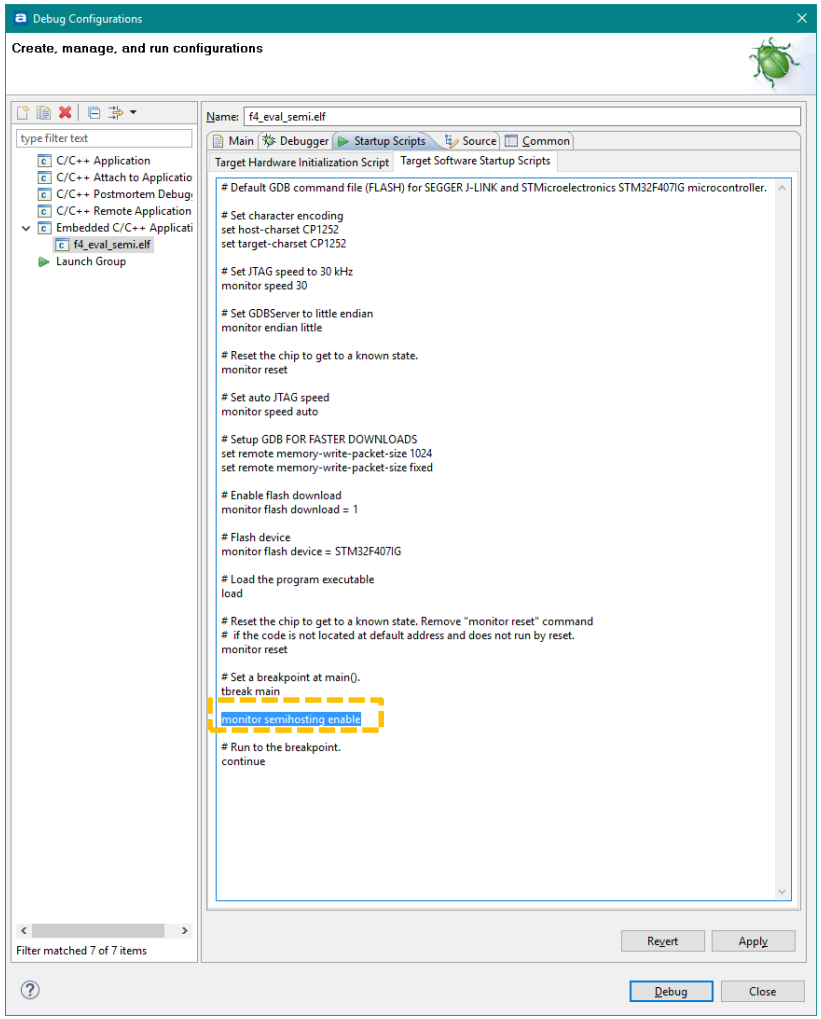
```
42
43 void initialise_monitor_handles();
44
45 int main(void)
46 {
47   initialise_monitor_handles();
48   int i = 0;
49
```

For C++ projects:

main.cpp

```
42
43 extern "C" void initialise_monitor_handles();
44
45 int main(void)
46 {
47   initialise_monitor_handles();
48   int i = 0;
49
```

Enable semihosting in the debug configuration

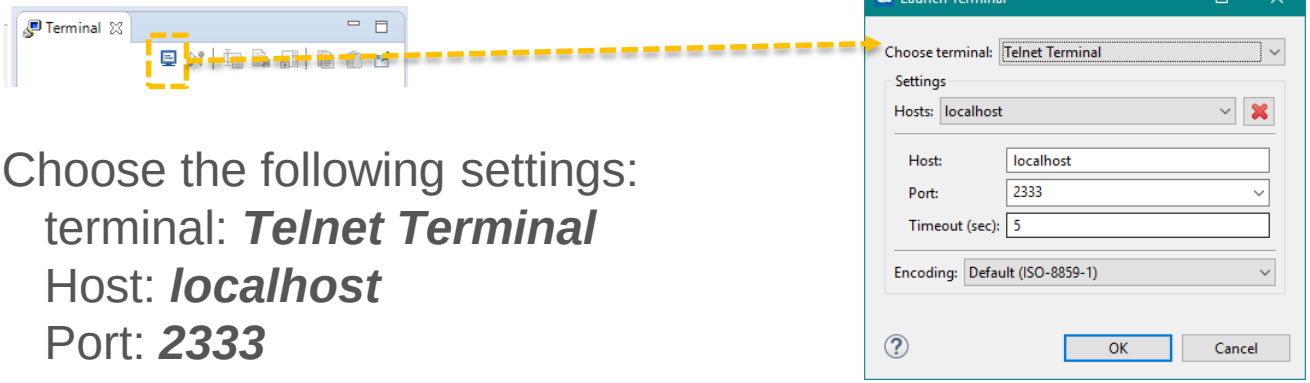


Open your debug configuration and add a line to enable semihosting for SEGGER J-Link:
monitor semihosting enabled

Launch debug session and setup terminal

1. Launch the debug session using the debug configuration previously modified.

2. Open the *Terminal* view and click the "*console*" icon to setup a new connection



3. Choose the following settings:
   terminal: ***Telnet Terminal***
   Host: ***localhost***
   Port: ***2333***

4. Press *run* on your application and your printfs should appear on *Terminal*