Introduction to ARM Debugger
ARM-ELF-INSIGHT Execution

```
$ arm-elf-insight
```

DREY 全:
Source Window

Program not running. Click on run icon to start.
Target Setting
Simulator Selection

Source Window

Target Selection

- Target: Simulator
- Options: GDBserver/Serial, GDBserver/TCP, Remote/Serial, Remote/TCP
- Port: /dev/com1

- Set breakpoint at 'main'
- Set breakpoint at 'exit'
- Set breakpoint at [ ]
- Display Download Dialog

OK  Cancel  Help
Open the Traced ELF File
Download the Debugged File into Simulator
Debugged File Comes Out
Breakpoint Setting
Run Program

Program is ready to run.
Display in Assembly Format

```
1 .file "sum.s"
2 .text
3 .align 2
4 .global main
5 .type main, %function
6 main:
7 @ args = 0, pretend = 0, frame = 12
8 @ frame_needed = 1, uses_anonymous_args = 0
9
-10 sub fp, ip, #4
-11 mov r9, #18
-12 str r3, [fp, #-20]
-13 mov r3, #20
-14 str r3, [fp, #-16]
-15 mov r3, #5
-16 str r3, [sp, #0]
-17 mov r3, #6
-18 str r3, [sp, #4]
-19 sub sp, fp, #12
-20 ldmfd sp, {fp, sp, pc}
-21 .size main, .-main
-22 .align 2
-23 .global sum6
-24 sub fp, sp, #12
-25 .size sum6, .-sum6
```

Program stopped at line 9
Memory Display
Modify Memory Range

<table>
<thead>
<tr>
<th>Address</th>
<th>0x7c00</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>C</td>
</tr>
<tr>
<td>0x00000000</td>
<td>0xe1a0c00d</td>
</tr>
<tr>
<td>0x00000010</td>
<td>0xe3a0300a</td>
</tr>
<tr>
<td>0x00000020</td>
<td>0xe3a03005</td>
</tr>
<tr>
<td>0x00000030</td>
<td>0xe3a00001</td>
</tr>
<tr>
<td>0x00000040</td>
<td>0xeb00001</td>
</tr>
<tr>
<td>0x00000050</td>
<td>0xe92dd810</td>
</tr>
<tr>
<td>0x00000060</td>
<td>0xe50b1020</td>
</tr>
<tr>
<td>0x00000070</td>
<td>0xe50b3018</td>
</tr>
</tbody>
</table>

Target is LITTLE endian