

Running
Step IntoF2
Step OverCtrl-F2
Step OutShift-F2
ContinueF3
StopF4
RestartCtrl-R
ReloadCtrl-Shift-L
File
Help

Registers
Refresh

r0	00000040
r1	00000050
r2	00000090
r3	ffffff00
r4	00001400
r5	00000000
r6	00000000
r7	00000000
r8	00000000
r9	00000000
r10	00000000
r11	00000000
r12	00000000
sp	00000000
lr	00000000
pc	00000014
cpsr	800001d3 NZCVI SVC
spsr	00000000 NZCVI ?

Registers
Call stack
Trace
Breakpoints
Watchpoints
Symbols
Counters

Settings

Number Display Options
Size: Word
Format: Hexadecimal
Memory words per row: 4

Editor (Ctrl-E)
Compile and Load (F5)
Language: ARMv7
untitled.s [changed since save]

```

1 //NAME= YASH KUMAR MISHRA
2 //REG NO = 2341013142
3 //SECTION = 23412F1
4 //LAB 4
5 //OBJECTIVE 1
6 .global _start
7 _start:
8     mov r0,#0x40
9     mov r1,#0x50
10    adds r2,r0,#0x50
11    subs r3,r0,#0x50
12    mul r4,r0,r1
13 my_exit: b my_exit
14

```

Editor (Ctrl-E)
Disassembly (Ctrl-D)
Memory (Ctrl-M)

Messages

Compiling...
Code and data loaded from ELF executable into memory. Total size is 24 bytes.

```

Assemble: arm-eabi-as -mfloat-abi=softfp -march=armv7-a -mcpu=cortex-a9 -mfpu=neon-fp16 --gdwarf2 -o work/asmgNj8wl.s.o work/asmgNj8wl.s
Link: arm-eabi-ld --script build_arm.ld -e _start -u _start -o work/asmgNj8wl.s.elf work/asmgNj8wl.s.o
Compile succeeded.

```

Running
Step IntoF2
Step OverCtrl-F2
Step OutShift-F2
ContinueF3
StopF4
RestartCtrl-R
ReloadCtrl-Shift-L
File
Help

Registers
Refresh

r0	10100010
r1	00000004
r2	00000001
r3	00000005
r4	00000003
r5	00000004
r6	00000000
r7	00000000
r8	00000000
r9	00000000
r10	00000000
r11	00000000
r12	00000000
sp	00000000
lr	00000000
pc	00000024
cpsr	200001d3 NZCVI SVC
spsr	00000000 NZCVI ?

Registers
Call stack
Trace
Breakpoints
Watchpoints
Symbols
Counters

Settings

Number Display Options
Size: Word
Format: Hexadecimal
Memory words per row: 4

Editor (Ctrl-E)
Compile and Load (F5)
Language: ARMv7
untitled.s [changed since save] [changed si

```

1 //NAME= YASH KUMAR MISHRA
2 //REG NO = 2341013142
3 //SECTION = 23412F1
4 //LAB 4
5 //OBJECTIVE 2
6 .global _start
7 _start:
8     LDR R0, =0X10100000
9     LDR R1, [R0], #4
10    LDR R2, [R0], #4
11    ADDS R3, R1, R2
12    STR R3, [R0], #4
13    SUBS R4, R1, R2
14    STR R4, [R0], #4
15    MUL R5, R1, R2
16    STR R5, [R0]
17 my_exit: b my_exit

```

Editor (Ctrl-E)
Disassembly (Ctrl-D)
Memory (Ctrl-M)

Messages

Compiling...
Code and data loaded from ELF executable into memory. Total size is 48 bytes.

```

Assemble: arm-eabi-as -mfloat-abi=softfp -march=armv7-a -mcpu=cortex-a9 -mfpu=neon-fp16 --gdwarf2 -o work/asmhqCChs.s.o work/asmhqCChs.s
Link: arm-eabi-ld --script build_arm.ld -e _start -u _start -o work/asmhqCChs.s.elf work/asmhqCChs.s.o
Compile succeeded.

```

Running

Step Into  
F2

Step Over  
Ctrl-F2

Step Out  
Shift-F2

Continue  
F3

Stop  
F4

Restart  
Ctrl-R

Reload  
Ctrl-Shift-L

File

Help

Registers

Refresh

r0	10100014
r1	00000004
r2	00000001
r3	00000000
r4	00000005
r5	00000005
r6	fffffffb
r7	00000000
r8	00000000
r9	00000000
r10	00000000
r11	00000000
r12	00000000
sp	00000000
lr	00000000
pc	0000002c
cpsr	400001d3

RegistersCall stackTraceBreakpointsWatchpointsSymbolsCounters

Settings

Number Display Options

Size: Word

Format: Hexadecimal

Memory words per row: 4

Editor (Ctrl-E)

Compile and Load (F5)

Language: ARMv7

untitled.s [changed since save] [changed since cor

```
1 //NAME= YASH KUMAR MISHRA
2 //REG NO = 2341013142
3 //SECTION = 23412F1
4 //LAB 4
5 //OBJECTIVE 3
6 .global _start
7 _start:
8     LDR R0, =0X10100000
9     LDR R1, [R0], #4
10    LDR R2, [R0], #4
11    ANDS R3, R2, R1
12    STR R3, [R0], #4
13    ORR R4, R2, R1
14    STR R4, [R0], #4
15    EOR R5, R2, R1
16    STR R5, [R0], #4
17    MVN R6, R1
18    STR R6, [R0]
19 my_exit: b my_exit
20
21
```

Editor (Ctrl-E)Disassembly (Ctrl-D)Memory (Ctrl-M)

Messages

Compiling...

Code and data loaded from ELF executable into memory. Total size is 56 bytes.

Assemble: arm-eabi-as -mfloat-abi=softfp -march=armv7-a -mcpu=cortex-a9 -mfpu=neon-fp16 --gdwarf2 -o work/asmbnRXmB.s.o work/asmbnRXmB.s

Link: arm-eabi-ld --script build\_arm.ld -e \_start -u \_start -o work/asmbnRXmB.s.elf work/asmbnRXmB.s.o

Compile succeeded.