

OSI Week4.b

1. Background: RISC-V assembly II

Assembler instructions with C expression operands:

asm(Template : OutputOperands : InputOperands)

a) Template: a string that is the template for the assembler code.

```
asm("lw ra,12(sp)");  
asm("ret");
```

b) OutputOperands: the C variables modified by the instructions in the Template.

```
void *sp;  
asm("mv %0, sp" : "=r"(sp));
```

c) InputOperands: C expressions read by the instructions in the Template.

```
void *sp = (void*)0x803fffc0;  
asm("mv sp,%0" :: "r"(sp));
```

[read more: <https://gcc.gnu.org/onlinedocs/gcc/Extended-Asm.html>]

2. Timer interrupt handler

```
void handler() {  
    CRITICAL("Got a timer interrupt!");  
    // (4) reset timer  
  
}  
  
int main() {  
    CRITICAL("This is a simple timer example");  
  
    // (1) register handler() as interrupt handler  
  
    // (2) set a timer  
  
    // (3) enable timer interrupt  
  
    while(1);  
}
```