Timer interrupts

[borrowed from Yunhao's
CS 4411/5411: Practicum in Operating Systems, 22fall]

CPU view: Core-local Interrupt (CLINT)

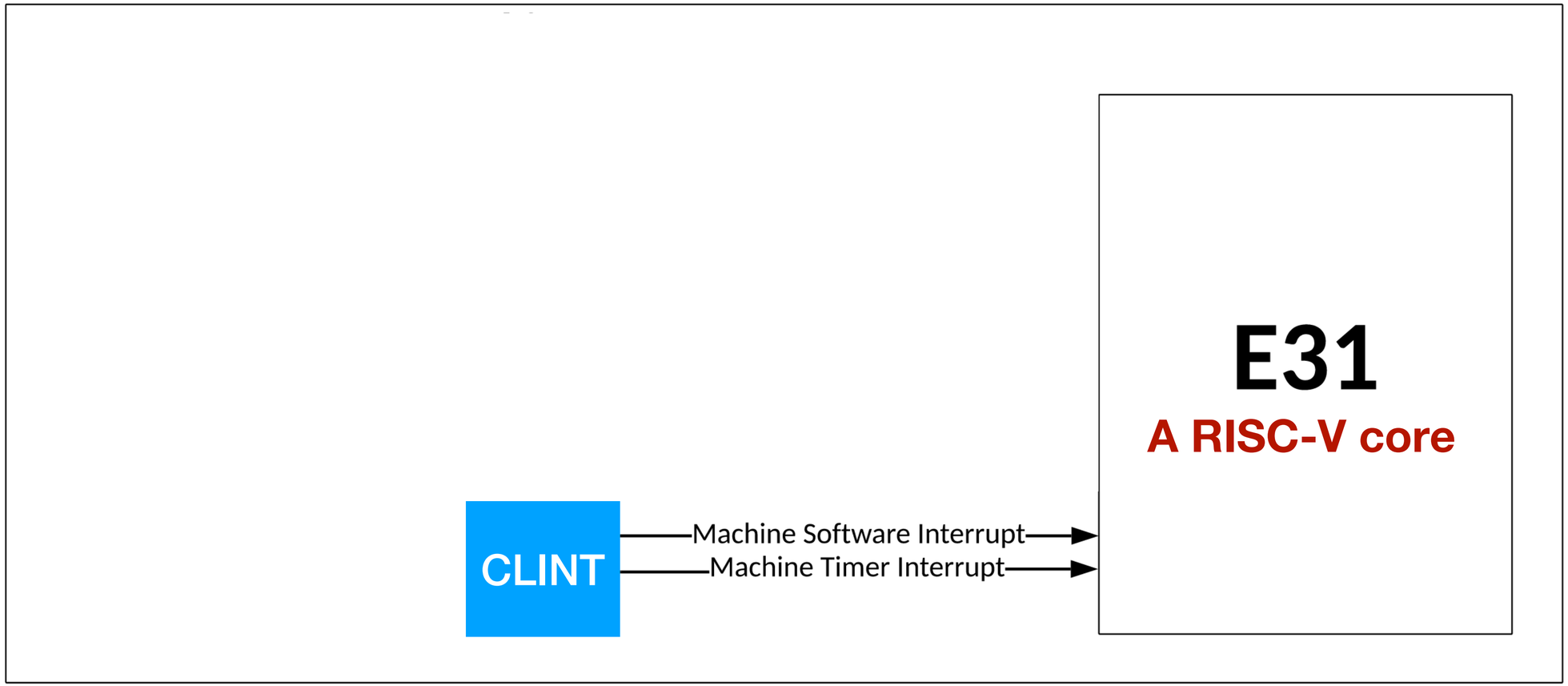


Figure 4 of Sifive FE310 manual

A programmer's view

```
void handler() {
    printf("Got a timer interrupt!");
    // (4) reset timer
}

int main() {
    // (1) register interrupt handler
    // (2) set a timer
    // (3) enable timer interrupt

    while(1);
}
```

The `mtvec` CSR

Value	Name	Description
0	Direct	All exceptions set `pc` to BASE.
1	Vectored	Asynchronous interrupts set `pc` to BASE+4×cause.
≥2	—	*Reserved*

Table 3.5: Encoding of `mtvec` MODE field.

RISC-V privileged spec

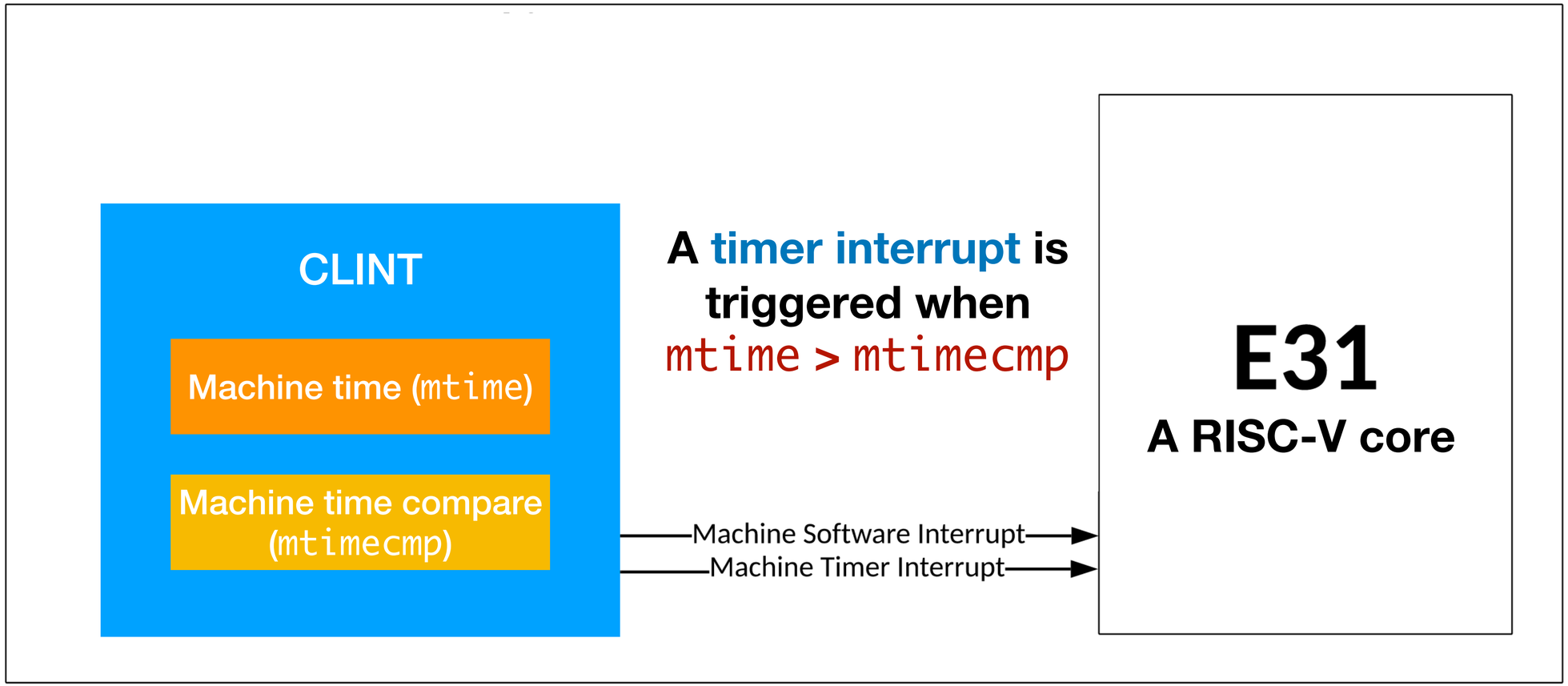
A programmer's view

```
void handler() {
    printf("Got a timer interrupt!");
    // (4) reset timer
}

int main() {
    // (1) register interrupt handler
    // (2) set a timer
    // (3) enable timer interrupt

    while(1);
}
```

The mtime and mtimecmp CSRs



Sifive FE310 manual

CLINT CSRs are memory-mapped

`mtimecmp_set()` writes 8 bytes to

`mtime_get()` reads 8 bytes from

Address	Width	Attr.	Description
0x2000000	4B	RW	msip for hart 0
0x2004008			Reserved
…			
0x200bff7			
0x2004000	8B	RW	mtimecmp for hart 0
0x2004008			Reserved
…			
0x200bff7			
0x200bff8	8B	RW	mtime
0x200c000			Reserved

Sifive FE310 manual

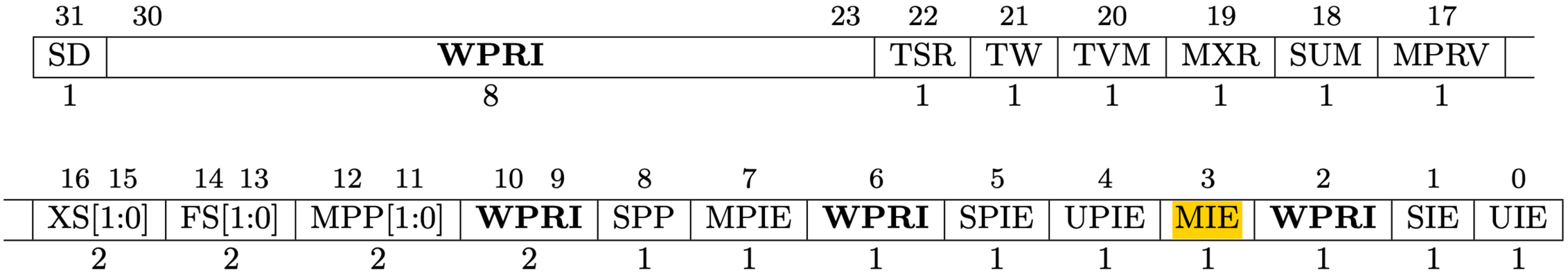
A programmer's view

```
void handler() {
    printf("Got a timer interrupt!");
    // (4) reset timer
}

int main() {
    // (1) register interrupt handler
    // (2) set a timer
    // (3) enable timer interrupt

    while(1);
}
```

The `mstatus` CSR



MIE stands for machine interrupt enable

RISC-V privileged spec

The `mie` CSR (not `mstatus.MIE`)

XLEN-1 12	11	10	9	8	7	6	5	4	3	2	1	0
WPRI	MEIE	**WPRI**	SEIE	UEIE	MTIE	**WPRI**	STIE	UTIE	MSIE	**WPRI**	SSIE	USIE
XLEN-12	1	1	1	1	1	1	1	1	1	1	1	1

MTIE stands for machine timer interrupt enable

RISC-V privileged spec

Next…

- two bummers in setting timers
- interrupts beyond CLINT
- RISC-V fine-grained control of interrupts
- how to know which interrupt is triggered?

a bummer: mtime rollover

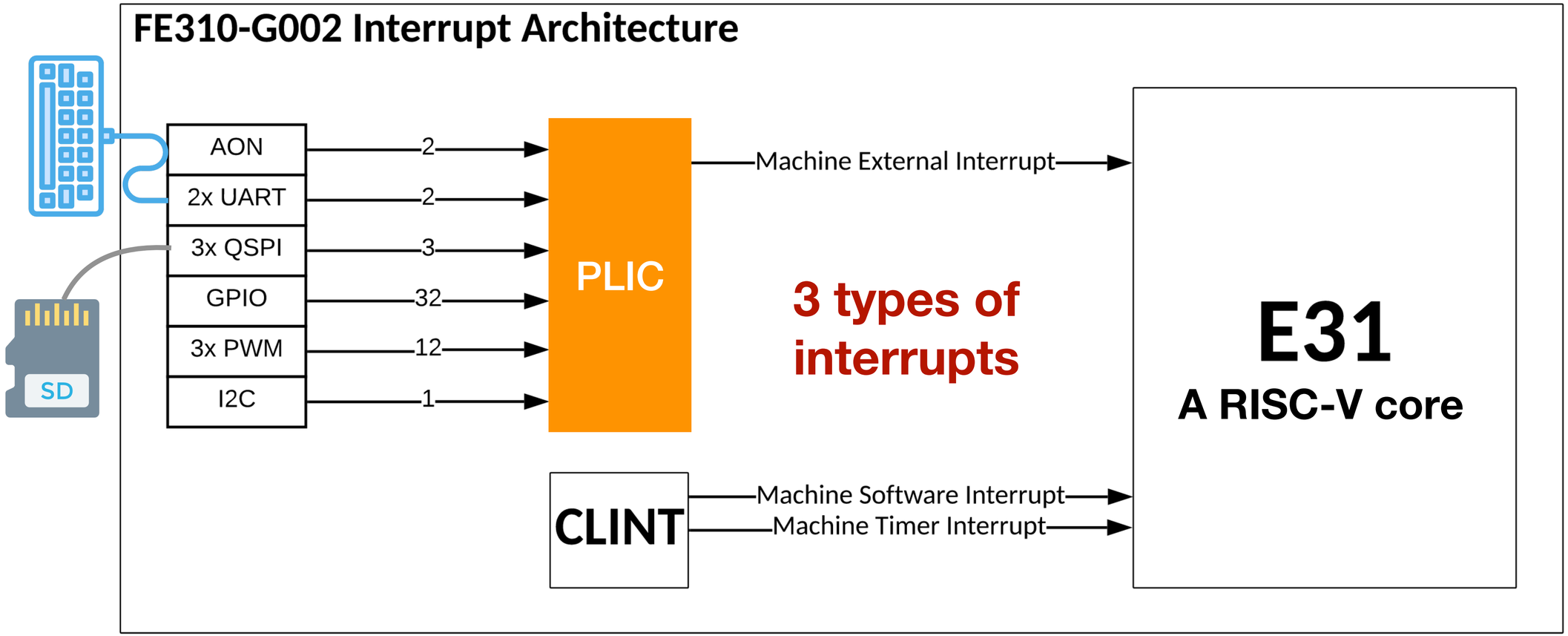
When reading the lower 4 bytes, mtime is
0x00000000 0xffffffff

When reading the higher 4 bytes, mtime is
0x00000001 0x00000000

Combine the two 0x00000001ffffffff is wrong!

There is another bummer: mtime overflow

Platform Interrupt Controller (PLIC)



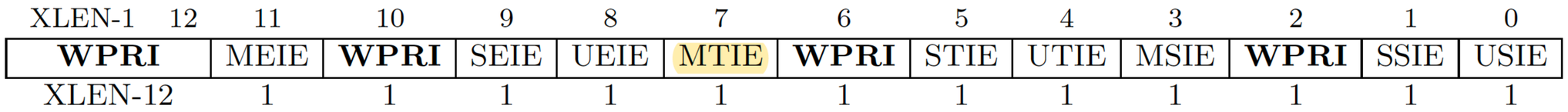
Sifive FE310 manual

`mie` provides fine-grained control

MTIE: machine **timer** interrupt enable

MEIE: machine **external** interrupt enable

MSIE: machine **software** interrupt enable



XLEN-1 … 12	11	10	9	8	7	6	5	4	3	2	1	0
WPRI	MEIE	**WPRI**	SEIE	UEIE	MTIE	**WPRI**	STIE	UTIE	MSIE	**WPRI**	SSIE	USIE
XLEN-12	1	1	1	1	1	1	1	1	1	1	1	1

Figure 3.12: Machine interrupt-enable register (`mie`).

Timer is interrupt #7

Interrupts

Exceptions

Interrupt	Exception Code	Description
1	0	*Reserved*
1	1	Supervisor software interrupt
1	2	*Reserved*
1	3	Machine software interrupt
1	4	*Reserved*
1	5	Supervisor timer interrupt
1	6	*Reserved*
1	7	Machine timer interrupt
1	8	*Reserved*
1	9	Supervisor external interrupt
1	10	*Reserved*
1	11	Machine external interrupt
1	12–15	*Reserved*
1	≥16	*Designated for platform use*
0	0	Instruction address misaligned
0	1	Instruction access fault
0	2	Illegal instruction
0	3	Breakpoint
0	4	Load address misaligned
0	5	Load access fault
0	6	Store/AMO address misaligned
0	7	Store/AMO access fault
0	8	Environment call from U-mode
0	9	Environment call from S-mode
0	10	*Reserved*
0	11	Environment call from M-mode
0	12	Instruction page fault
0	13	Load page fault
0	14	*Reserved*
0	15	Store/AMO page fault
0	16–23	*Reserved*
0	24–31	*Designated for custom use*
0	32–47	*Reserved*
0	48–63	*Designated for custom use*
0	≥64	*Reserved*

System calls?

Interrupts

Exceptions

Interrupt	Exception Code	Description
1	0	*Reserved*
1	1	Supervisor software interrupt
1	2	*Reserved*
1	3	Machine software interrupt
1	4	*Reserved*
1	5	Supervisor timer interrupt
1	6	*Reserved*
1	7	Machine timer interrupt
1	8	*Reserved*
1	9	Supervisor external interrupt
1	10	*Reserved*
1	11	Machine external interrupt
1	12–15	*Reserved*
1	≥16	*Designated for platform use*
0	0	Instruction address misaligned
0	1	Instruction access fault
0	2	Illegal instruction
0	3	Breakpoint
0	4	Load address misaligned
0	5	Load access fault
0	6	Store/AMO address misaligned
0	7	Store/AMO access fault
0	8	Environment call from U-mode
0	9	Environment call from S-mode
0	10	*Reserved*
0	11	Environment call from M-mode
0	12	Instruction page fault
0	13	Load page fault
0	14	*Reserved*
0	15	Store/AMO page fault
0	16–23	*Reserved*
0	24–31	*Designated for custom use*
0	32–47	*Reserved*
0	48–63	*Designated for custom use*
0	≥64	*Reserved*