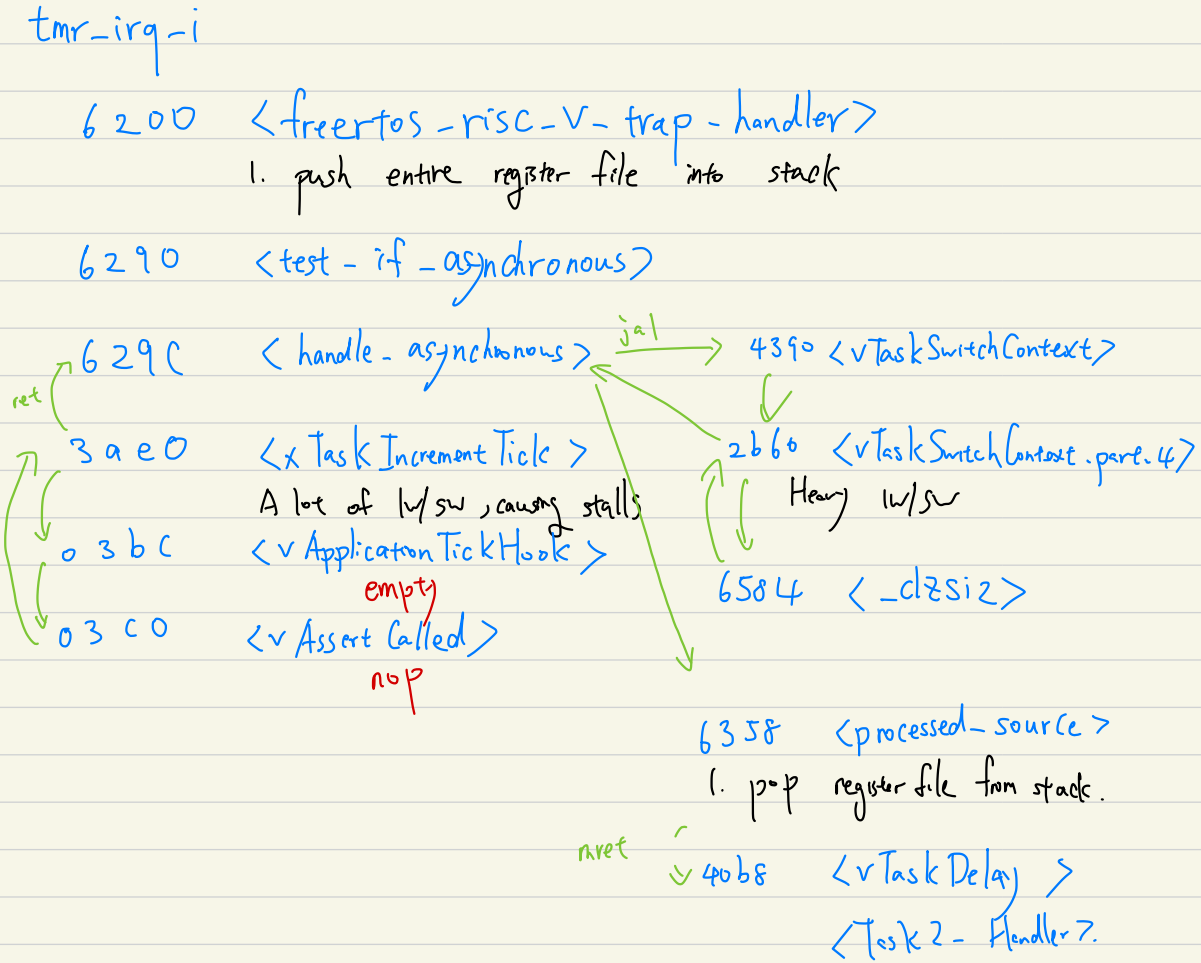
tmr_irq_i
6200 <freertos_risc_v_trap_handler>
1. push entire register file into stack
6290 <test_if_asynchronous>
629C <handle_asynchronous>
jal
4390 <vTaskSwitchContext>
ret
3ae0 <xTaskIncrementTick>
A lot of lw/sw, causing stalls
2b60 <vTaskSwitchContext.part.4>
Heavy lw/sw
03bc <vApplicationTickHook>
empty
6584 <_clzsi2>
03c0 <vAssertCalled>
nop
6358 <processed_source>
1. pop register file from stack.
mret
40b8 <vTaskDelay>
<Task2_Handler>.

<freertos-risc-v-trap_handler>

store entire register file & machine status CSR (mstatus)

additional regs

load mcause - cause of interrupt
load mepc - interrupted PC value

<test-if asynchronous>

skip interrupt handle if it's synchronous interrupt, shown by MSB of mcause.

<handle asynchronous>

add and set value for mtime & mtimecmp. (for next timer interrupt)

<xTaskIncrementTick>

default: pdFALSE;
if return pdTRUE: task in ready que has higher priority than task now, PendSV interrupt do context switch.

xTickCount ++;

<vTaskSwitchContext>