Lecture3

CORTEX M_RESET SEQUENCE

By: Abdelrahman matarawy

Switch Between Privileged and Unprivileged:

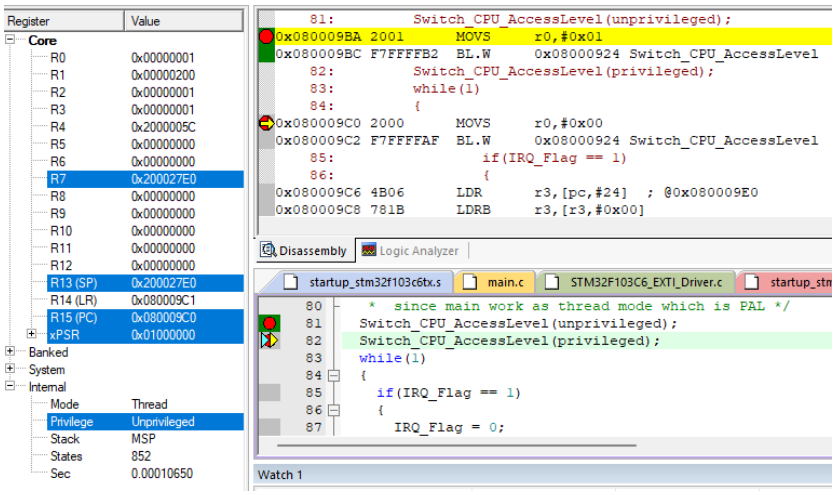
- Code:

```c
enum CPU_AccessLevel{
    privileged,
    unprivileged
};

void Switch_CPU_AccessLevel(enum CPU_AccessLevel LEVEL)
{
    switch(LEVEL)
    {
    case privileged:
        /* Clear Bit 0 in CONTROL register */
        __asm("MRS R3, CONTROL      \n\t"   // Store value of Control register to R3
              "LSR R3, R3, #0x01  \n\t"   // Shift R3 to right to remove bit 0
              "LSL R3, R3, #0x01  \n\t"   // Shift R3 to left to clear bit 0
              "MSR CONTROL, R3");         // load R3 value to Control Register after edit value of bit 0
        break;
    case unprivileged:
        /* Set Bit 0 in CONTROL register */
        __asm("MRS R3, CONTROL \n\t"        // Store value of Control register to R3
              "orr R3, R3, #0x01 \n\t"    // Set bit 0
              "MSR CONTROL, R3");         // load R3 value to Control Register after edit value of bit 0
        break;
    }
}
```

```c
int main(void)
{
    clk_init();
    /* Set Configuration of EXTI */
    EXTI_Config_t EXTI_Conf;  // Make Variable of EXTI_PinConfi
    EXTI_Conf.EXTI_PIN = EXTI_LINE9_PB9; // Set line 9, GPIOB,
    EXTI_Conf.Trigger_Case = EXTI_Trigger_RASING; // Specifies
    EXTI_Conf.IRQ_Case =EXTI_IRQ_Enable; // Enable Interrupt Ma
    EXTI_Conf.P_IRQ_CallBack = EXTI_CallBack; // Make Call Back

    MCAL_EXTI_GPIO_Init(&EXTI_Conf);
    IRQ_Flag = 1;

    /* When Try to test our function first make it unprivileged
     *  since main work as thread mode which is PAL */
    Switch_CPU_AccessLevel(unprivileged);
    Switch_CPU_AccessLevel(privileged);
    while(1)
    {
        if(IRQ_Flag == 1)
        {
            IRQ_Flag = 0;
        }

    }
}
```

```
1008    d009        beq.n   800094c <Switc
1009    e00f        b.n 800095a <Switch_CP
1010    f3ef 8314   mrs r3, CONTROL
1011    ea4f 0353   mov.w   r3, r3, lsr #1
1012    ea4f 0343   mov.w   r3, r3, lsl #1
1013    f383 8814   msr CONTROL, r3
1014    e006        b.n 800095a <Switch_CP
1015    f3ef 8314   mrs r3, CONTROL
1016    f043 0301   orr.w   r3, r3, #1
1017    f383 8814   msr CONTROL, r3
1018    bf00        nop
1019    bf00        nop
1020    370c        adds    r7, #12
1021    46bd        mov sp, r7
1022    bc80        pop {r7}
1023    4770        bx  lr
1024
1025<clk_init>:
1026    b480        push    {r7}
1027    af00        add r7, sp, #0
1028    4b07        ldr r3, [pc, #28]   ;
1029    699b        ldr r3, [r3, #24]
1030    4a06        ldr r2, [pc, #24]   ;
1031    f043 0308   orr.w   r3, r3, #8
1032    6193        str r3, [r2, #24]
```

- Simulation:



➢ Note:

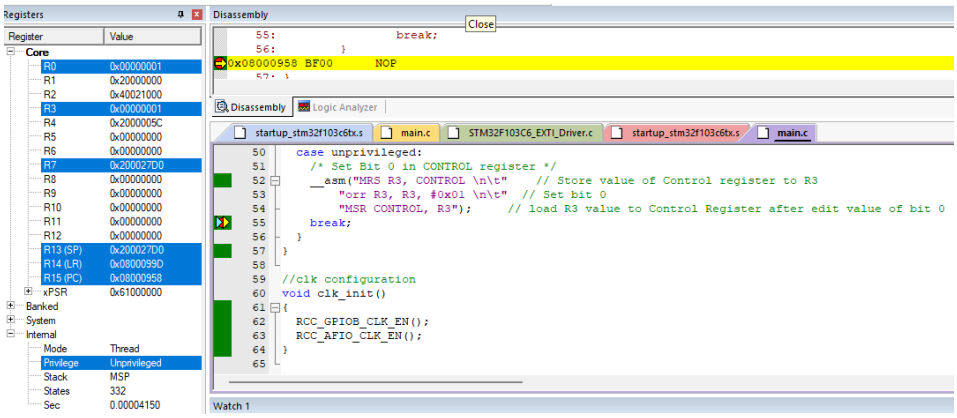
Not allowed to write in CONTROL register when you are unprivileged, so we handle it in ISR as handler mode work as PAL to it have access to write on Control.

What would happen if we switched to unprivileged before configuring NVIC?

- Code:

```
65
66  int main(void)
67  {
68      clk_init();
69
70      /* Test what will happen if change NPAL before init the EXTI */
71      Switch_CPU_AccessLevel(unprivileged);
72
73      /* Set Configuration of EXTI */
74      EXTI_Config_t EXTI_Conf;  // Make Variable of EXTI_PinConfig_t structure
75      EXTI_Conf.EXTI_PIN = EXTI_LINE9_PB9; // Set line 9, GPIOB, Pin9 an EXTI
76      EXTI_Conf.Trigger_Case = EXTI_Trigger_RASING; // Specifies Rising Trigger
77      EXTI_Conf.IRQ_Case =EXTI_IRQ_Enable; // Enable Interrupt Mask
78      EXTI_Conf.P_IRQ_CallBack = EXTI_CallBack; // Make Call Back Function point to ISR
79
80      MCAL_EXTI_GPIO_Init(&EXTI_Conf);
81      IRQ_Flag = 1;
82
83      /* When Try to test our function first make it unprivileged
84       *  since main work as thread mode which is PAL */
85      //Switch_CPU_AccessLevel(unprivileged);
86      Switch_CPU_AccessLevel(privileged);
```

- Simulation:



➢ Note:

The Stack branch to strange address, we find that it loops in hard fault handler because not allow to access NVIC during unprivileged.

Switch Between them using ISR:

- Code:

```c
void EXTI_CallBack(void)
{
    IRQ_Flag = 1;
    Switch_CPU_AccessLevel(privileged);
}

int main(void)
{
    clk_init();

    /* Set Configuration of EXTI */
    EXTI_Config_t EXTI_Conf;  // Make Variable of EXTI_PinConfig_t structure
    EXTI_Conf.EXTI_PIN = EXTI_LINE9_PB9; // Set line 9, GPIOB, Pin9 an EXTI
    EXTI_Conf.Trigger_Case = EXTI_Trigger_RASING; // Specifies Rising Trigger
    EXTI_Conf.IRQ_Case =EXTI_IRQ_Enable; // Enable Interrupt Mask
    EXTI_Conf.P_IRQ_CallBack = EXTI_CallBack; // Make Call Back Function point to ISR

    MCAL_EXTI_GPIO_Init(&EXTI_Conf);
    IRQ_Flag = 1;

    /* When Try to test our function first make it unprivileged
     *  since main work as thread mode which is PAL */
    Switch_CPU_AccessLevel(unprivileged);
    //Switch_CPU_AccessLevel(privileged);
    while(1)
```

- Simulation: