异常重映射方案搭建

一.搭建步骤

- 重新编写异常向量表
- 设置异常处理函数，主要重新设置irq和两种中止模式。
- 通过arm a9的cp15协处理器指令重新设置异常向量表基地址

二.主流程代码

1.重新编写异常向量表

```
start_redirect:
    b       reset
    ldr     pc, Undefined_Addr
    ldr     pc, SVC_Addr
    ldr     pc, Prefetch_Addr
    ldr     pc, Abort_Addr
    nop     @ Reserved for secure montor calls
    ldr     pc, IRQ_Addr
    ldr     pc, FIQ_Addr


Undefined_Addr:  .word     CPU_ARM_ExceptUndefInstrHndlr
SVC_Addr:        .word     CPU_ARM_ExceptSwiHndlr
Prefetch_Addr:   .word     CPU_ARM_ExceptPrefetchAbortHndlr
Abort_Addr:      .word     CPU_ARM_ExceptDataAbortHndlr
IRQ_Addr:        .word     CPU_ARM_ExceptIrqHndlr
FIQ_Addr:        .word     CPU_ARM_ExceptFiqHndlr
```

2.设置异常处理函数

```
@@ the 3 situations do not need to be handled.
@@
reset:
    b .

CPU_ARM_ExceptSwiHndlr:
    b .

CPU_ARM_ExceptFiqHndlr:
    b .
```

```
CPU_ARM_ExceptUndefInstrHndlr:
    b .

@@ the next 4 situations need to be handled.
@@
CPU_ARM_ExceptPrefetchAbortHndlr:
    @@ save
    sub     lr, lr, #8          @ adjust lr_prefetch to return
    srsdb   sp!, #0x13          @ store reture state to svc stack
    cps     #0x13               @ switch to svc mode to handle exception
    push    {r0-r3, r12}        @ caller saved registers
    @@ need to send a interrupt
    bl      prefetch_handler    @ bl somewhere to handle exception
    @@ restore
    pop     {r0-r3, r12}        @ caller saved registers
    rfeia   sp!                 @ Return from Exception from svc stack

CPU_ARM_ExceptDataAbortHndlr:
    @@ save
    sub     lr, lr, #4          @ adjust lr_data to return
    srsdb   sp!, #0x13          @ store reture state to svc stack
    cps     #0x13               @ switch to svc mode to handle exception
    push    {r0-r3, r12}        @ caller saved registers
    @@ need to send a interrupt
    bl      data_handler        @ bl somewhere to handle exception
    @@ restore
    pop     {r0-r3, r12}        @ caller saved registers
    rfeia   sp!                 @ Return from Exception from svc stack

CPU_ARM_ExceptIrqHndlr:
    @@ save
    sub     lr, lr, #4          @ adjust lr_irq to return
    srsdb   sp!, #0x13          @ store reture state to svc stack
    cps     #0x13               @ switch to svc mode to handle exception
    push    {r0-r3, r12}        @ caller saved registers
    @@ identify source, handle irq, clear source
    bl      identify_source
    cpsie   i
    bl      irq_handler         @ bl somewhere to handle exception
    cpsid   i
    @@ restore
    pop     {r0-r3, r12}        @ caller saved registers
    rfeia   sp!                 @ Return from Exception from svc stack
```

以上主要设置了发生prefetch abort和data abort异常时通知fpga，另外irq异常重新设置，包括新的中断函数注册和中断目标权限等的设置。

3.重新设置异常向量表基地址

```
@@ it is for the c caller to call
```

```
@@ to redirect the  vector table.
@@ need to be call after mmu enable.
reinit_vector:
    @@ save
    push    {r4-r11, lr}        @ callee saved registers
    @@ map the vector table to 0x00000000, for after relocation
    mrc     p15, 0, r0, c1, c0, 0
    bic     r0, #(0x1<<13)
    mcr     p15, 0, r0, c1, c0, 0
    @@ remap the vector table to start_redirect
    ldr     r0, =start_redirect
    mcr     p15, 0, r0, c12, c0, 0
    isb
    @@ restore
    pop     {r4-r11, pc}        @
```