Task 6

Write a test to change behaviour of a locked pmp region using smepmp extension in spike (Smepmp task)

a. Pre-requisites

- Read RISCV privileged architecture spec for PMP
- Read about Smepmp extension

b. Write an assembly test.

- Start your test in M mode. Configure a locked pmp region for read, write and execute permissions.
- What two things happen when you configure a locked pmp region??
- Now, try to change permissions of that region to only read from read,write and execute. How it can be achieved using smepmp extension

➢ Solution:

Github Link:

https://github.com/BilalAli10x/Riscv_Arch_Test/tree/main/BilalAli_Task6

Test Description

This test verifies:

1. Behavior of a locked PMP region (L=1).
2. That locked PMP entries:
    - Become immutable.
    - Apply to M-mode.
3. How Smepmp (mseccfg.RLB) allows modification of locked PMP entries.
4. Runtime permission change from RWX -> Read-only.
5. Behavior difference:
    - With SMEPMP disabled
    - With SMEPMP enabled

The test runs on Spike and validates behavior using CSR readback and trap detection.

Q): What Two Things Happen When PMP Region is Locked?

From Privileged Spec:

When L = 1 for a PMP entry:

1) Entry Becomes Immutable

- Writes to:
    - pmpcfgX
    - pmpaddrX
- Are ignored.
- The entry cannot be modified until reset.

2) Entry Applies to M-mode

Normally:

- PMP does NOT restrict M-mode.

But when L = 1:

- The region also applies to Machine mode.
- M-mode accesses are checked against that PMP entry.

This is explicitly defined in:

Privileged Spec – PMP Section 3.7.1 (Page no 60)

Why Locked PMP Cannot Be Modified Normally

- **Without Smepmp:**
  csrw pmpcfg0, new_value
- **If entry has L=1:**
  -> Write is silently ignored.
- **Readback:**
  csrr t1, pmpcfg0
  Will show old value.

How Smepmp Solves This

From Smepmp extension:

- **The mseccfg CSR introduces:**
  -> RLB (Rule Locking Bypass) Bit
- **If:**
  mseccfg.RLB = 1
- **Then:**
  Locked PMP entries may be modified.
  Lock protection is bypassed.
- **BUT -> Important rule from spec:**
  If any PMP entry has L=1 while RLB=0, then RLB becomes read-only 0.
- **So:**
  -> RLB must be set **before locking behavior permanently blocks it**.

Implementation Overview

Step 1: Configure PMP Regions

Two TOR regions configured:

Region	Mode	Permissions	Lock
0	TOR	RWX	0
1	TOR	RWX	1

Configuration written:

- pmpcfg0 = 0x00008F0F
- **Region 1:**
  L=1, A=TOR, X=1, W=1, R=1

Step 2: Test Locked Region (Initial RWX)

Operations performed in M-mode:

- Store -> Success
- Load -> Success
- Execute -> Success

Because region is RWX.

Step 3: Attempt to Change Locked Region to Read-Only

New configuration attempted:

pmpcfg0 = 0x0000890F

Region 1 becomes:

Now behavior depends on SMEPMP.

Case 1 – WITHOUT SMEPMP

- (#define SMEPMP commented)
- (#define NO_SMEPMP uncommented)

Expected Behavior:

- Write to locked PMP entry is ignored.
- Readback shows old configuration.
- RWX permissions remain active.
- No traps occur.

CSR Check in Code

- csrr t1, pmpcfg0
- beq t1, t4, fail_cfg_change

If configuration changed -> FAIL

Since it should NOT change.

Access Results

Operation	**Result**
Write	Success
Read	Success
Execute	Success

Final Output:

PASS (gp = 1)

Spike Log Snapshot:

```
core   0: 0x80000078 (0x0054a423) sw      t0, 8(s1)
core   0: 3 0x80000078 (0x0054a423) mem 0x80001008 0xdeadbeef
core   0: 0x8000007c (0x0084a283) lw      t0, 8(s1)
core   0: 3 0x8000007c (0x0084a283) x5  0xdeadbeef mem 0x80001008
core   0: 0x80000080 (0x000480e7) jalr    s1
core   0: 3 0x80000080 (0x000480e7) x1  0x80000084
core   0: 0x80001000 (0x00000013) nop
core   0: 3 0x80001000 (0x00000013)
core   0: 0x80001004 (0x00008067) ret
core   0: 3 0x80001004 (0x00008067)
core   0: 0x80000084 (0x1240006f) j       pc + 0x124
core   0: 3 0x80000084 (0x1240006f)
core   0: 0x800001a8 (0x00100193) li      gp, 1
core   0: 3 0x800001a8 (0x00100193) x3  0x00000001
core   0: 0x800001ac (0x00002297) auipc   t0, 0x2
core   0: 3 0x800001ac (0x00002297) x5  0x800021ac
core   0: 0x800001b0 (0xe432aa23) sw      gp, -428(t0)
```

Case 2 – WITH SMEPMP (#define SMEPMP enabled)

- (#define SMEPMP commented)
- (#define NO_SMEPMP uncommented)

Before configuring PMP:

- li t0, (1 << 2)
- csrw mseccfg, t0

This sets:

- mseccfg.RLB = 1

What RLB Does

Rule Lock Bypass allows:

- Modification of locked PMP entries.

Expected Behavior

- Configuration change succeeds.
- Readback matches new value.
- Region becomes Read-only.
- Write and Execute cause traps.

Access Results

Operation	Result
Write	Store Access Fault
Read	Success
Execute	Instruction Access Fault

Trap handler confirms:

- mcause = 1 -> Instruction fault
- mcause = 7 -> Store fault

Final Output

PASS (gp = 1)

Spike Log Snapshot:

```
80   core   0: exception trap_store_access_fault, epc 0x80000080
81   core   0:           tval 0x80001008
82   core   0: >>>>  trap_handler
```

```
109   core   0: exception trap_instruction_access_fault, epc 0x80001000
110   core   0:           tval 0x80001000
111   core   0: >>>>  trap_handler
112   core   0: 0x80000100 (0x342022f3) csrr    t0, mcause
```

Reference to Specification

RISC-V Privileged Architecture Specification

- **Section 3.1.15** - mcause
  - Instruction page fault = 12
  - Load page fault = 13
  - Store/AMO page fault = 15
- **Section 3.7:PMP Section** - Physical Memory Protection CSRs
- **Section 3.7.1:PMP Section** - Address matching and L-bit behavior
- **SMEPMP Spec Chapter 2:** Proposal behavior of bit 2