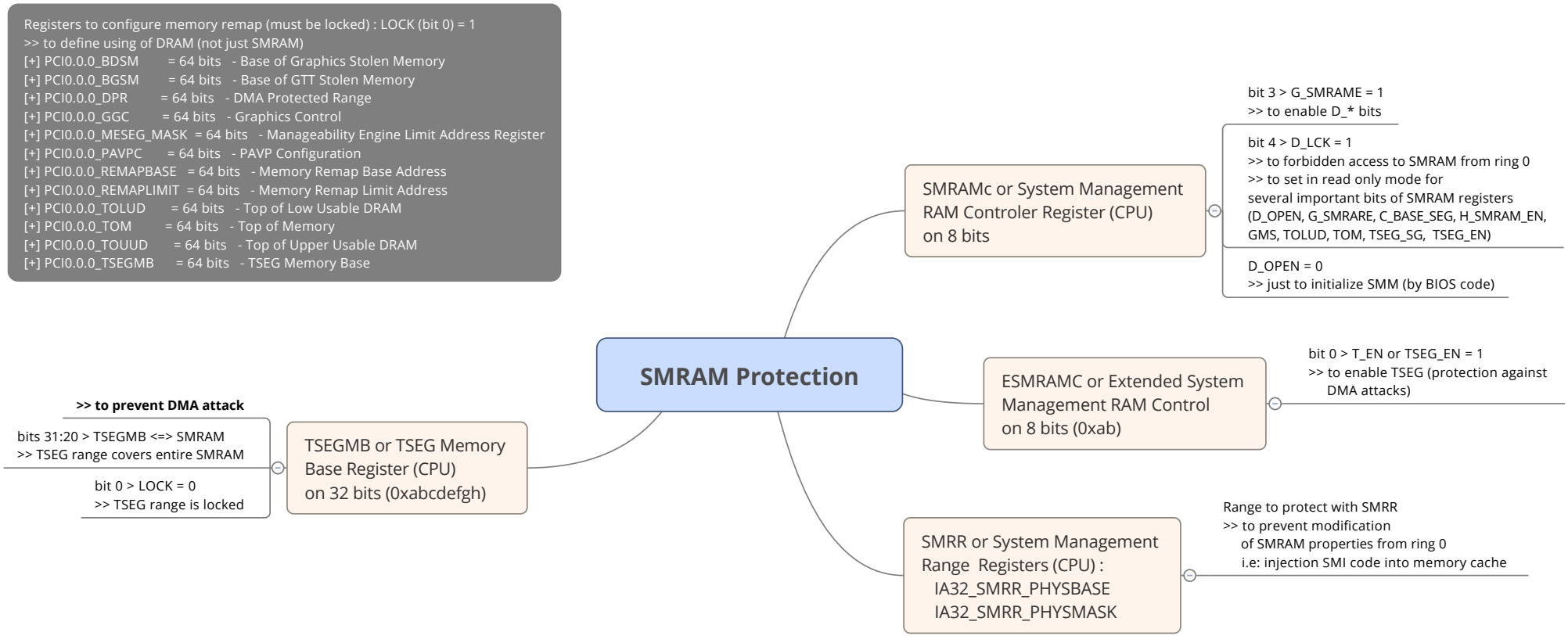
Registers to configure memory remap (must be locked) : LOCK (bit 0) = 1
>> to define using of DRAM (not just SMRAM)
[+] PCI0.0.0_BDSM = 64 bits - Base of Graphics Stolen Memory
[+] PCI0.0.0_BGSM = 64 bits - Base of GTT Stolen Memory
[+] PCI0.0.0_DPR = 64 bits - DMA Protected Range
[+] PCI0.0.0_GGC = 64 bits - Graphics Control
[+] PCI0.0.0_MESEG_MASK = 64 bits - Manageability Engine Limit Address Register
[+] PCI0.0.0_PAVPC = 64 bits - PAVP Configuration
[+] PCI0.0.0_REMAPBASE = 64 bits - Memory Remap Base Address
[+] PCI0.0.0_REMAPLIMIT = 64 bits - Memory Remap Limit Address
[+] PCI0.0.0_TOLUD = 64 bits - Top of Low Usable DRAM
[+] PCI0.0.0_TOM = 64 bits - Top of Memory
[+] PCI0.0.0_TOUUD = 64 bits - Top of Upper Usable DRAM
[+] PCI0.0.0_TSEGMB = 64 bits - TSEG Memory Base
SMRAM Protection
SMRAMc or System Management RAM Controler Register (CPU) on 8 bits
bit 3 > G_SMRAME = 1
>> to enable D_* bits
bit 4 > D_LCK = 1
>> to forbidden access to SMRAM from ring 0
>> to set in read only mode for
several important bits of SMRAM registers
(D_OPEN, G_SMRARE, C_BASE_SEG, H_SMRAM_EN,
GMS, TOLUD, TOM, TSEG_SG, TSEG_EN)
D_OPEN = 0
>> just to initialize SMM (by BIOS code)
ESMRAMC or Extended System Management RAM Control on 8 bits (0xab)
bit 0 > T_EN or TSEG_EN = 1
>> to enable TSEG (protection against DMA attacks)
SMRR or System Management Range Registers (CPU) :
IA32_SMRR_PHYSBASE
IA32_SMRR_PHYSMASK
Range to protect with SMRR
>> to prevent modification
of SMRAM properties from ring 0
i.e: injection SMI code into memory cache
TSEGMB or TSEG Memory Base Register (CPU) on 32 bits (0xabcdefgh)
>> to prevent DMA attack
bits 31:20 > TSEGMB <=> SMRAM
>> TSEG range covers entire SMRAM
bit 0 > LOCK = 0
>> TSEG range is locked