

# Towards (reasonably) trustworthy x86 laptops

Joanna Rutkowska

Invisible Things Lab & Qubes OS Project

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



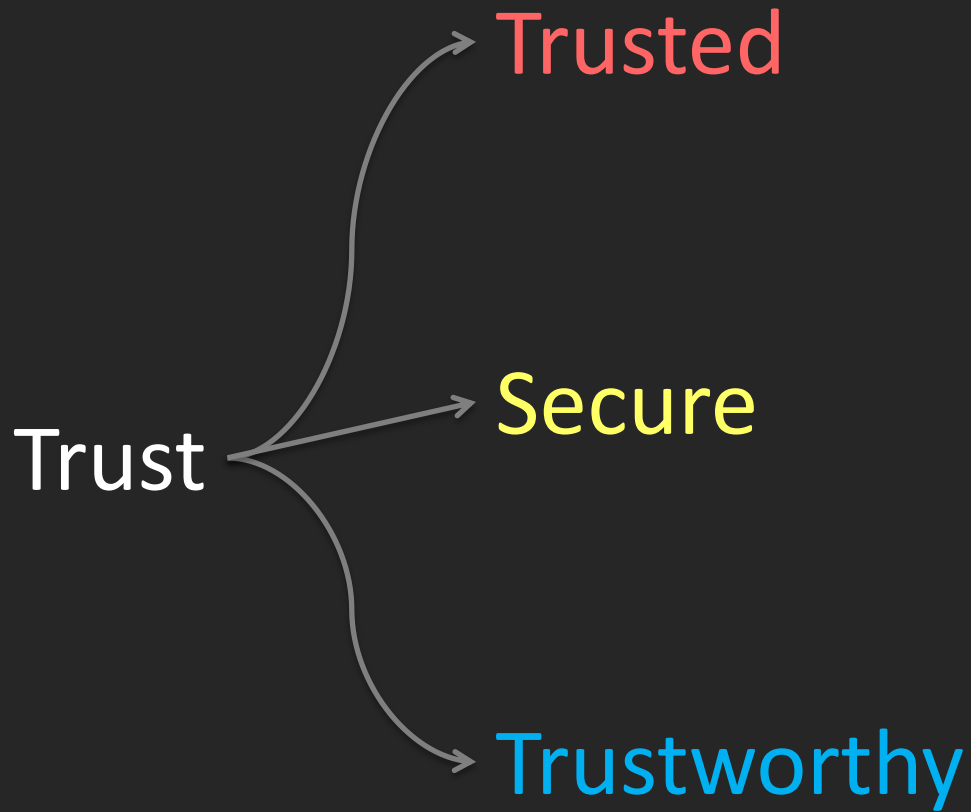
Personal computers are extensions of our  
brains...



...yet they are **insecure** and **untrustworthy** :(

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494





Apps

GnuPG, Tor,  
Pond, SSH,  
OpenVPN,  
LUKS,...



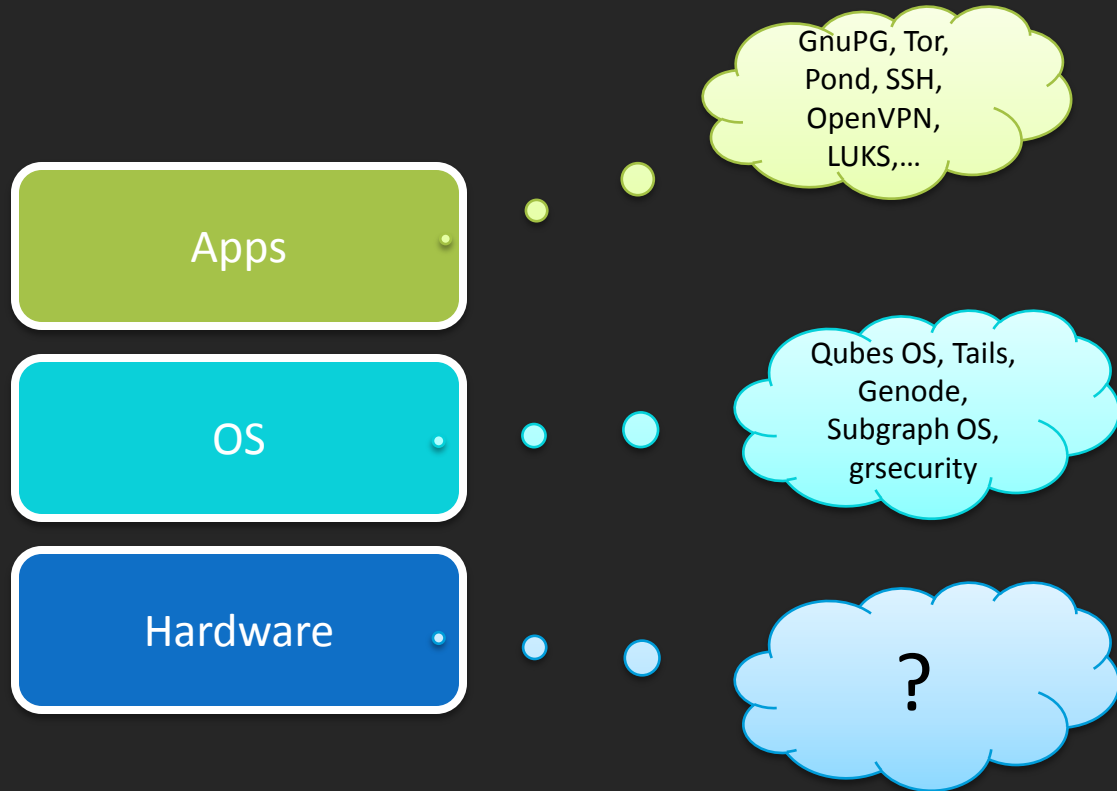
Apps

OS

GnuPG, Tor,  
Pond, SSH,  
OpenVPN,  
LUKS,...

Qubes OS, Tails,  
Genode,  
Subgraph OS,  
grsecurity



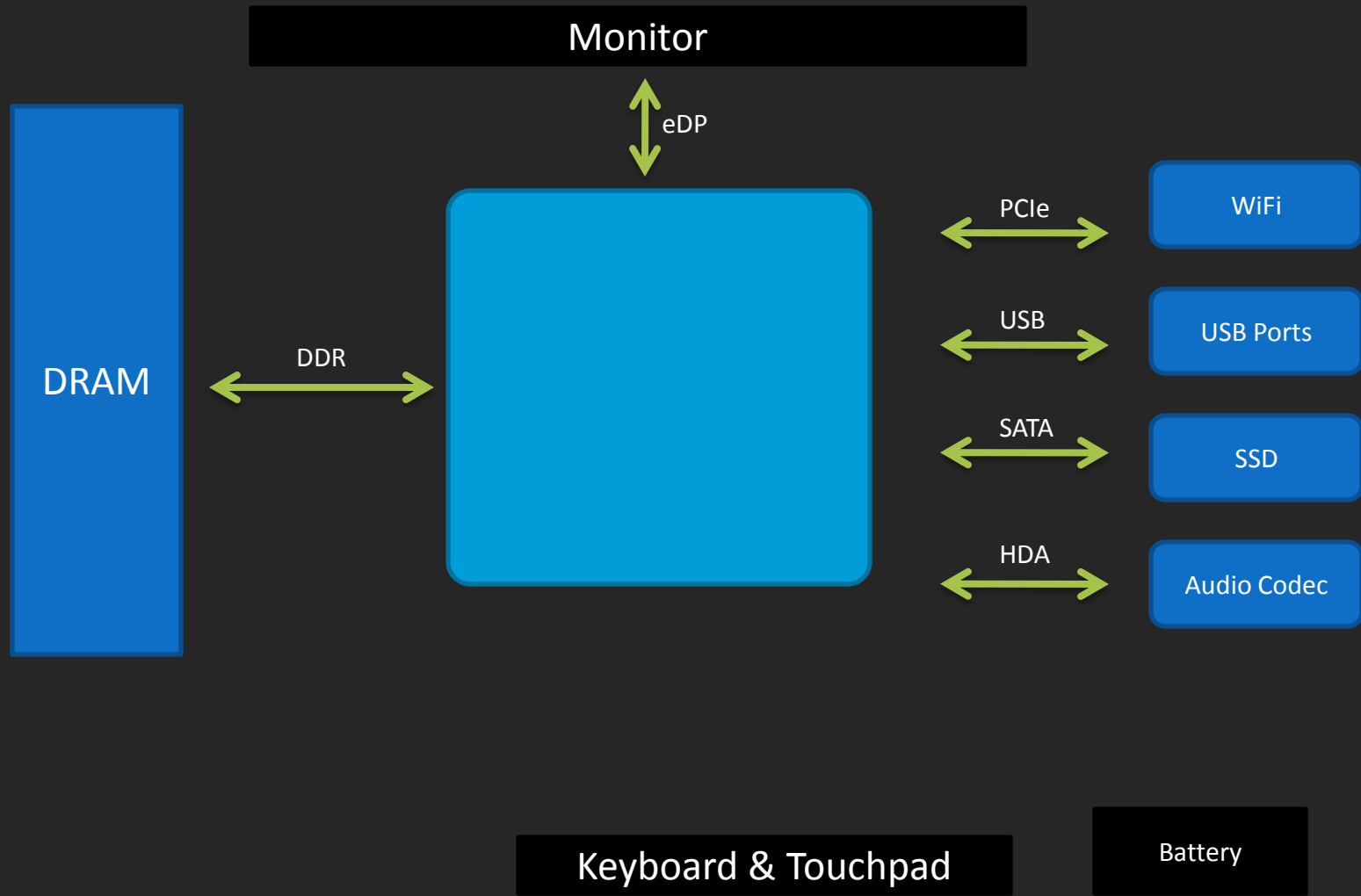


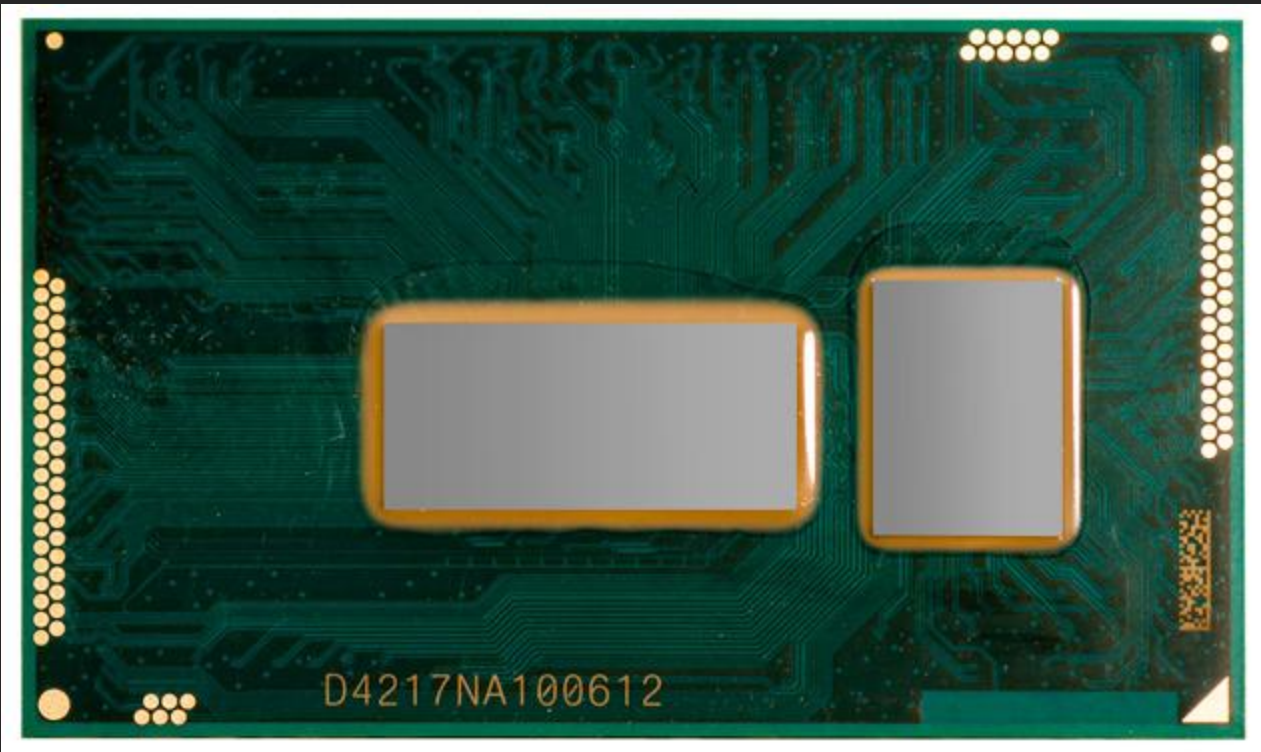
# What's wrong with (Intel) x86 laptops?

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



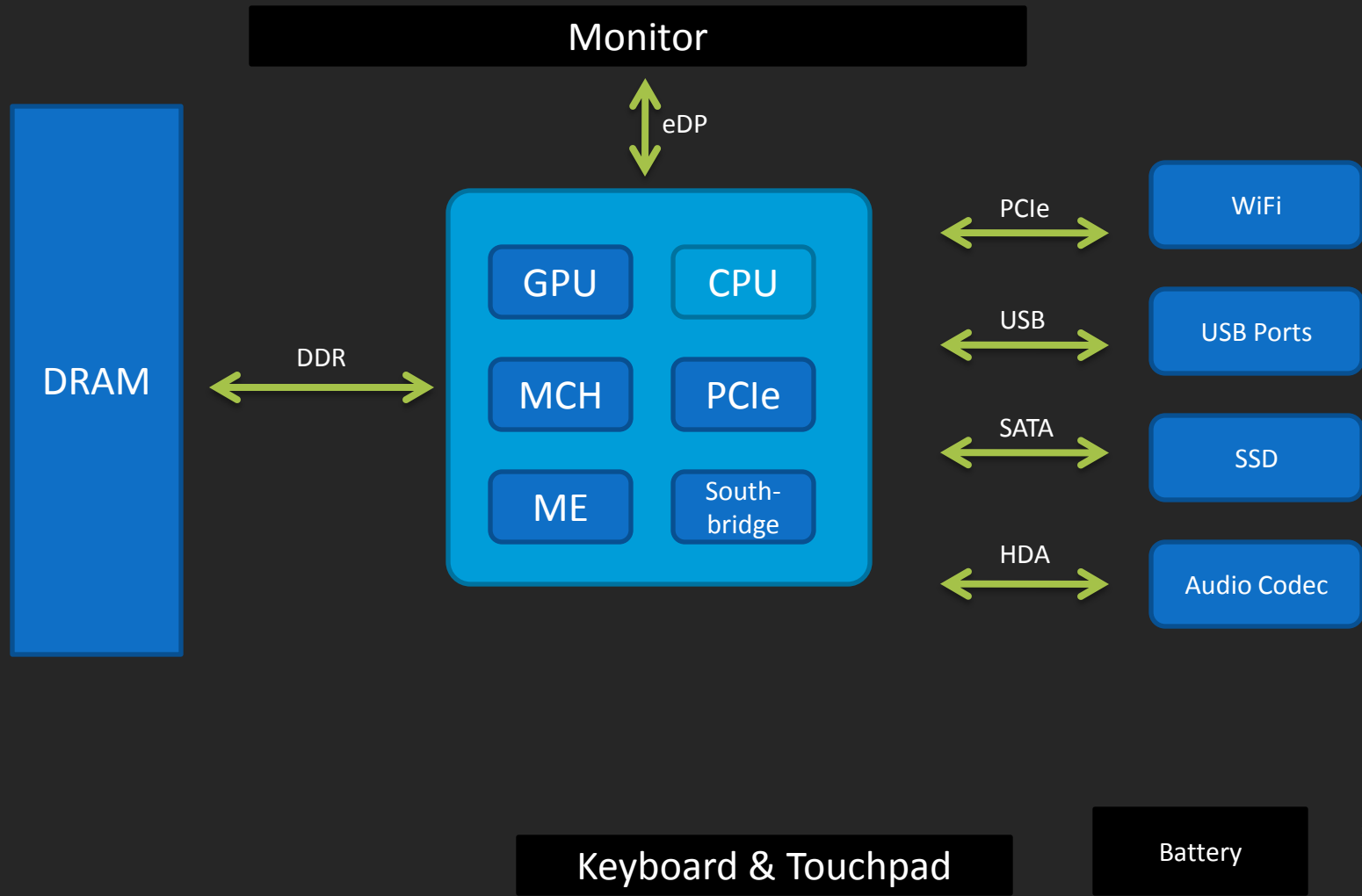


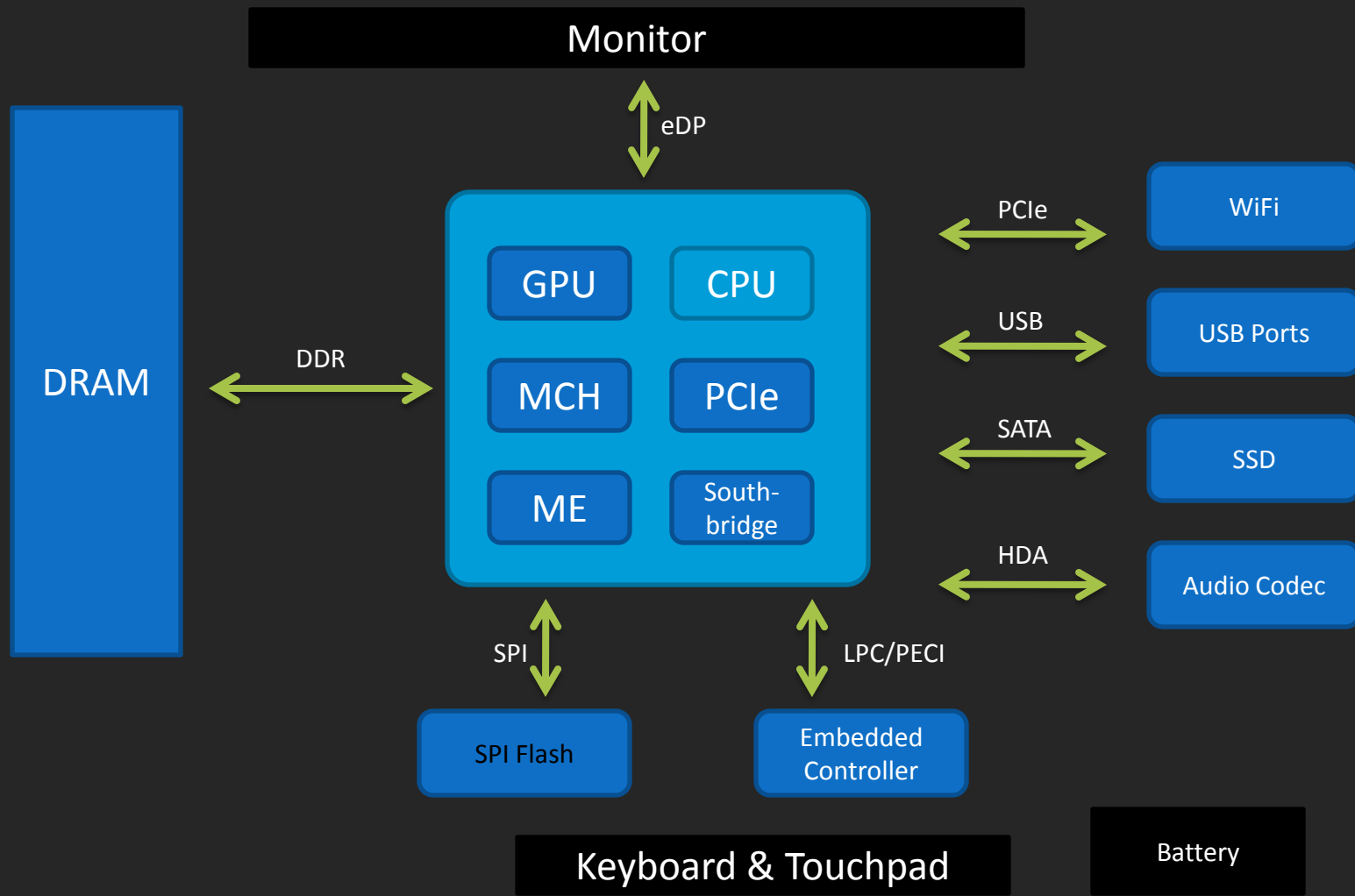




427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494





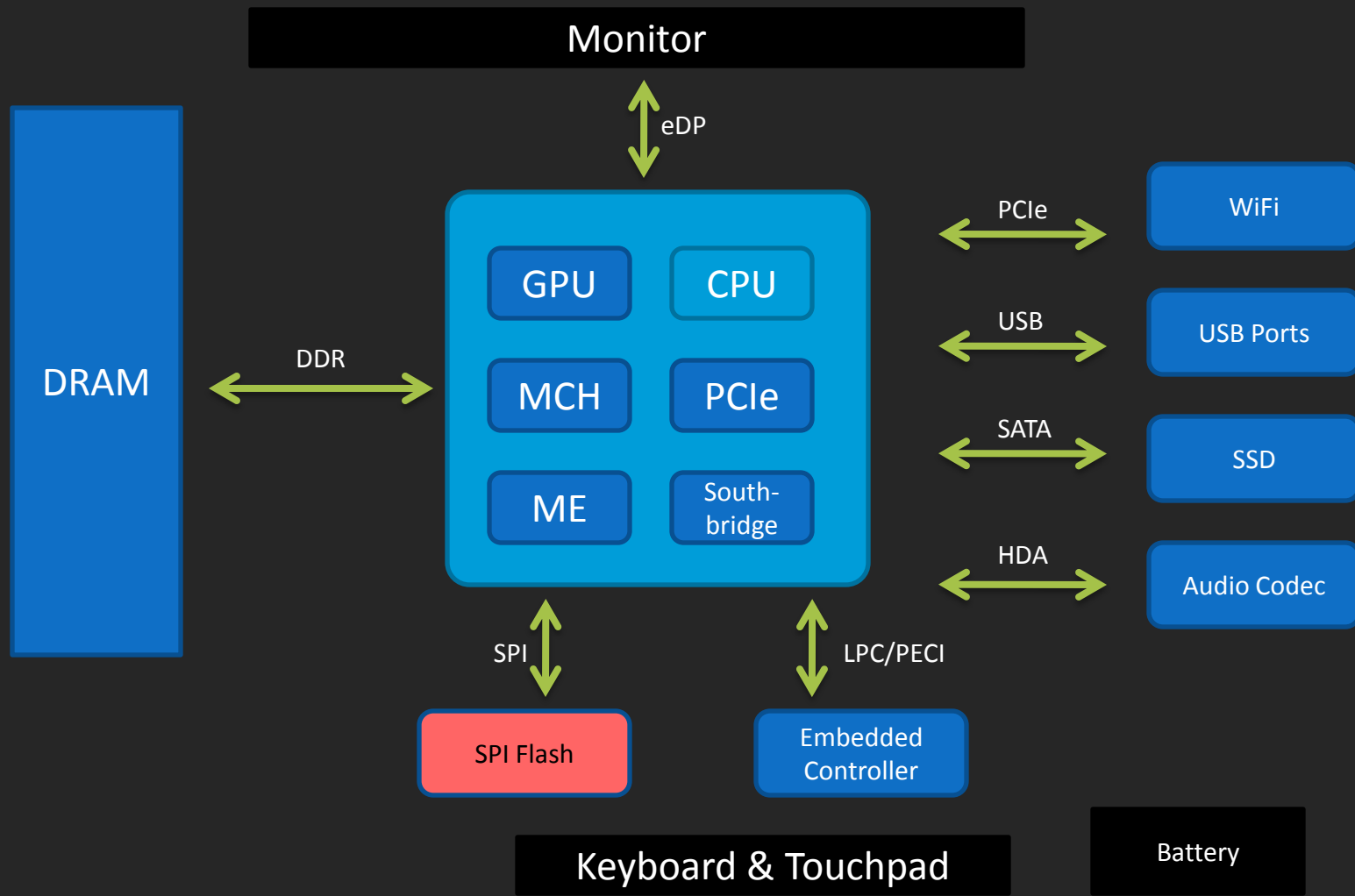


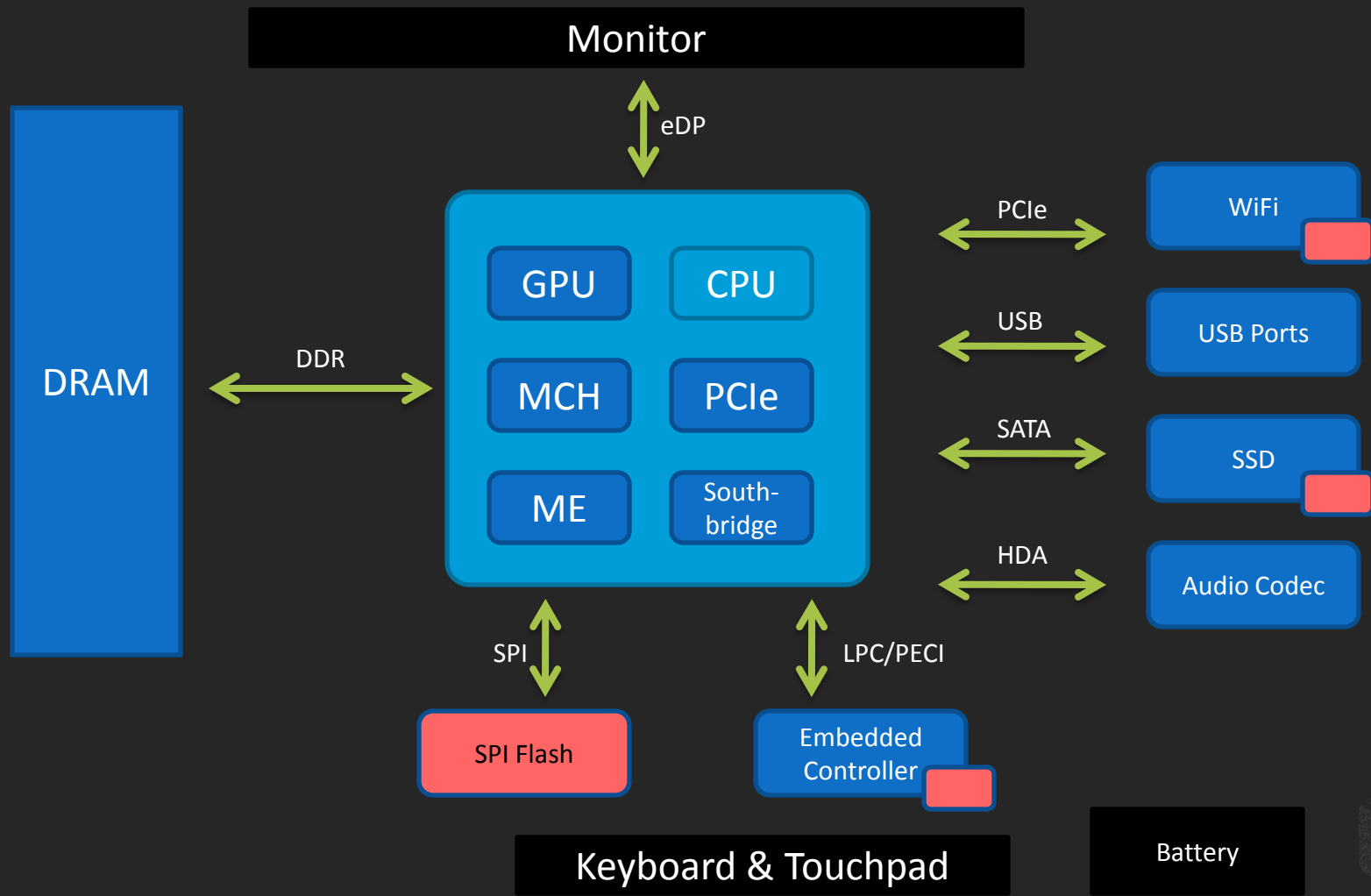
# Boot security?



# Malicious peripherals?

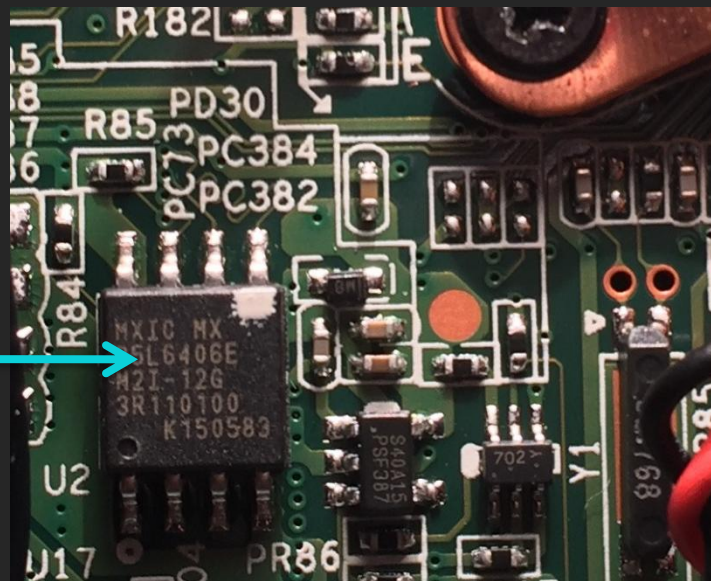








BIOS  
ME  
Config  
GbE  
?



What firmware *actually* is there?

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494

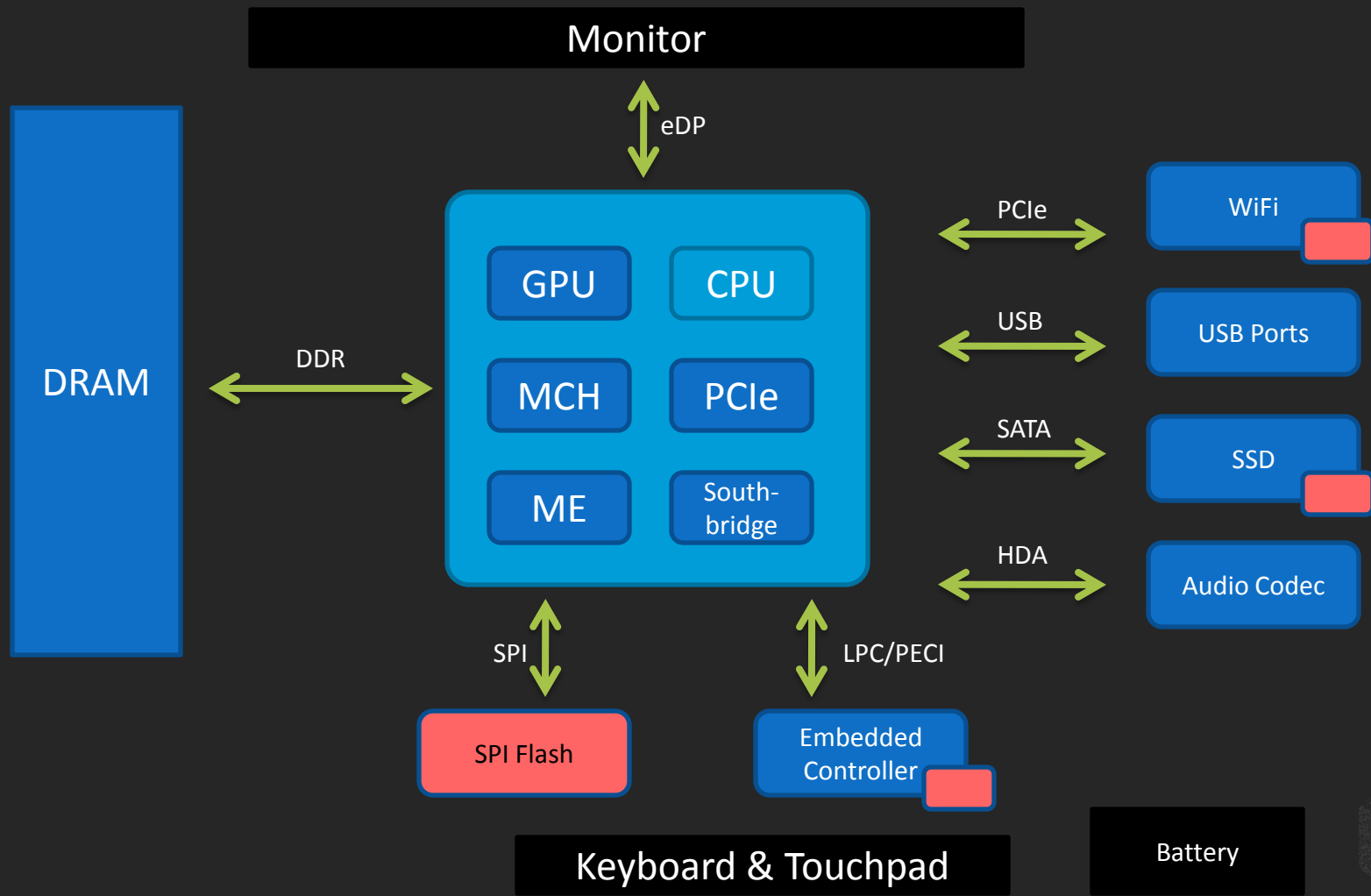


Can we enforce read-only'iness?



Can I upload *my own* firmware?





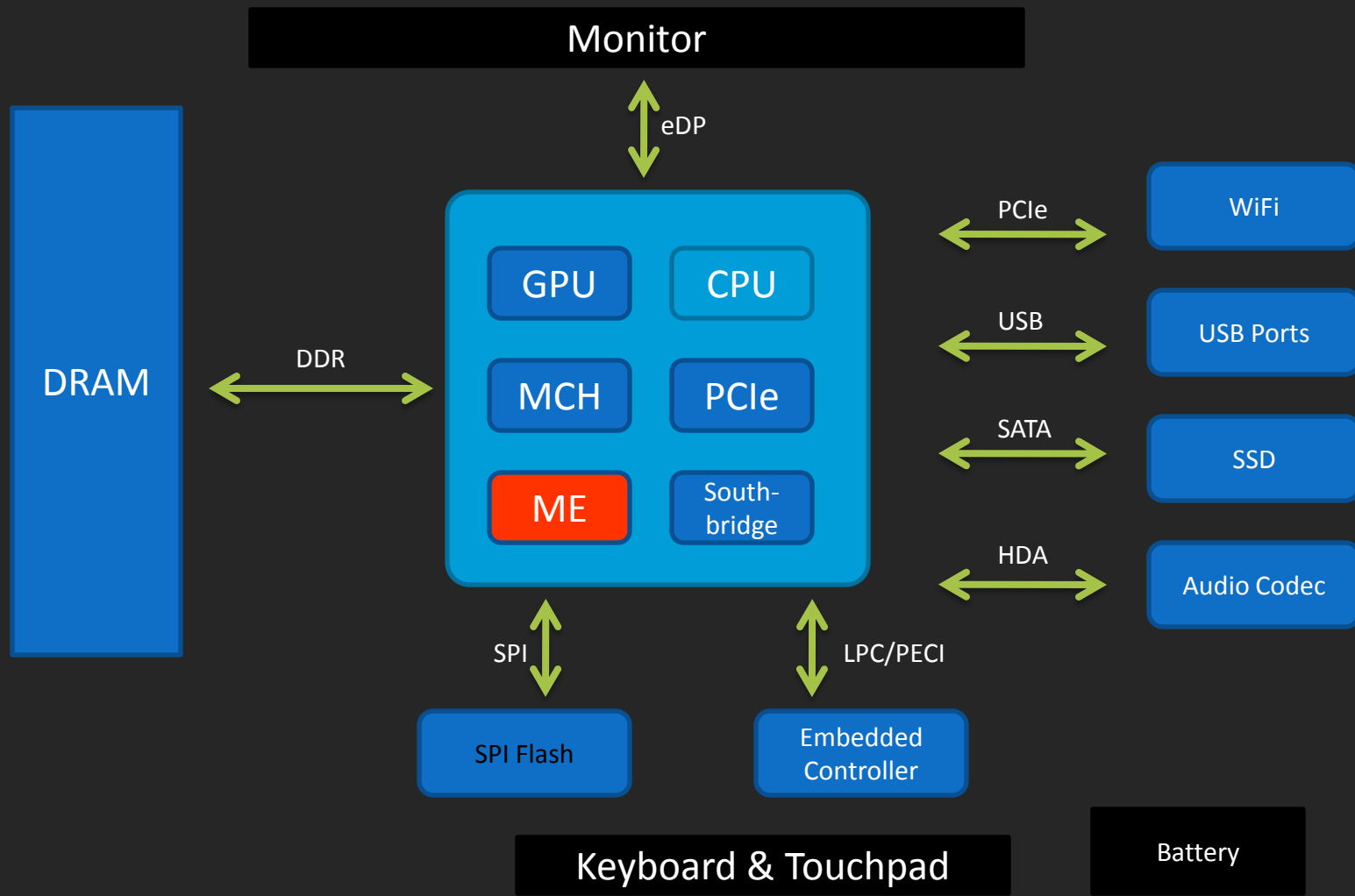
TPM? TXT? UEFI Secure Boot?



427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



# Intel ME





Intel  
ME

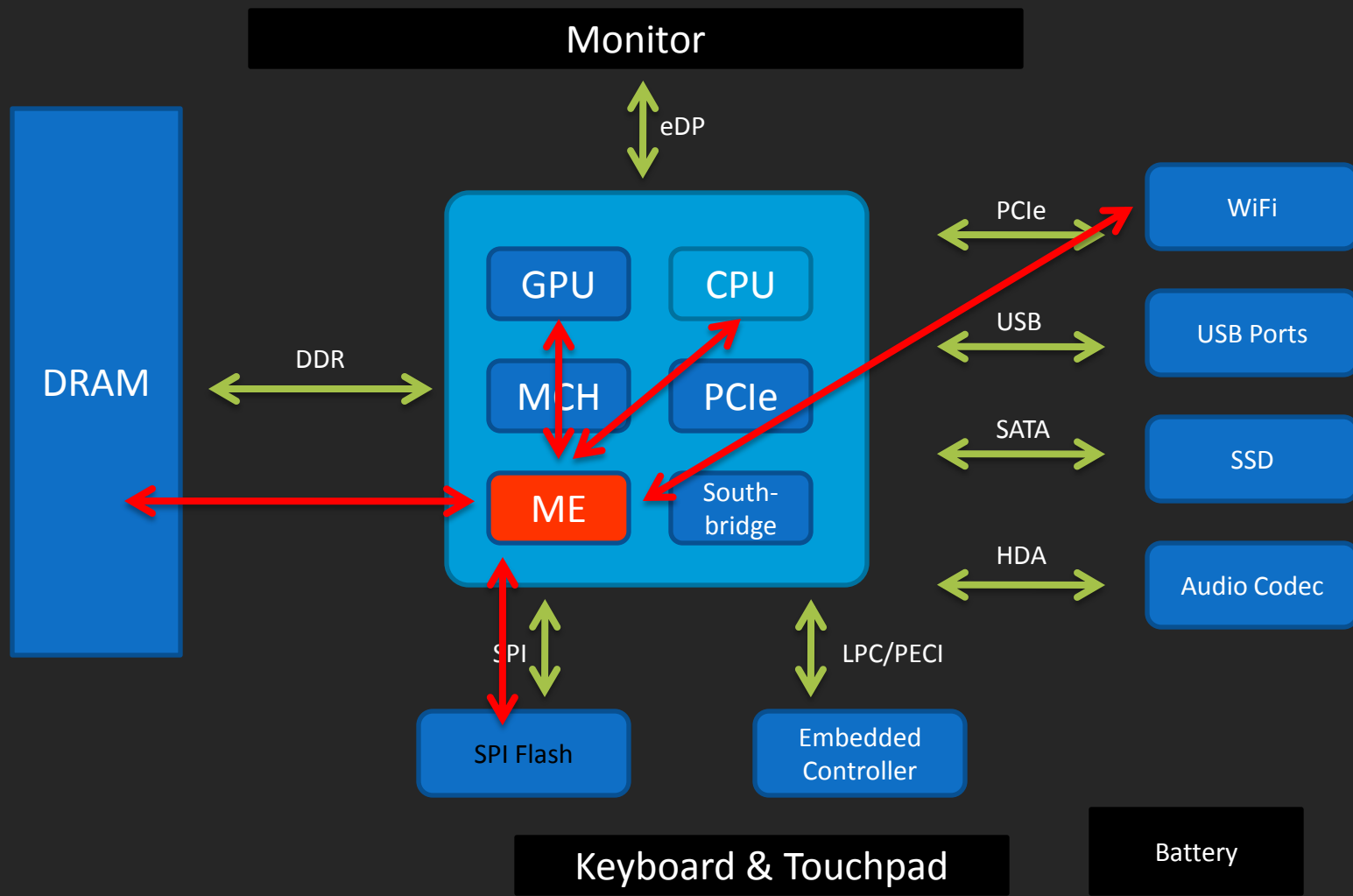
Backdooring & rootkiting

“Zombification” of  
personal computing





# Backdooring & rootkiting infrastructure



# “Zombification”

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494

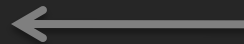
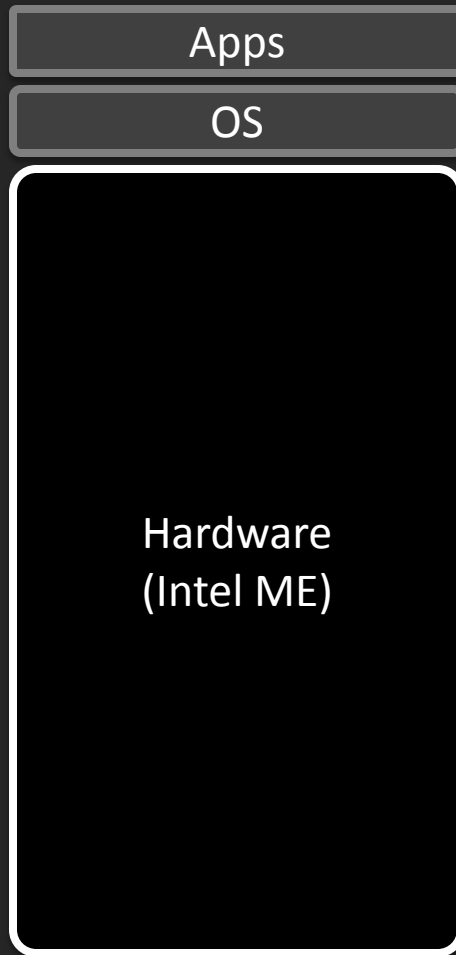




Apps

OS

Hardware



Zombies?

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



Can we disable Intel ME?  
Can we control what code it runs?  
Can we see what code is runs?



VT-d? TXT?





# A lost war?

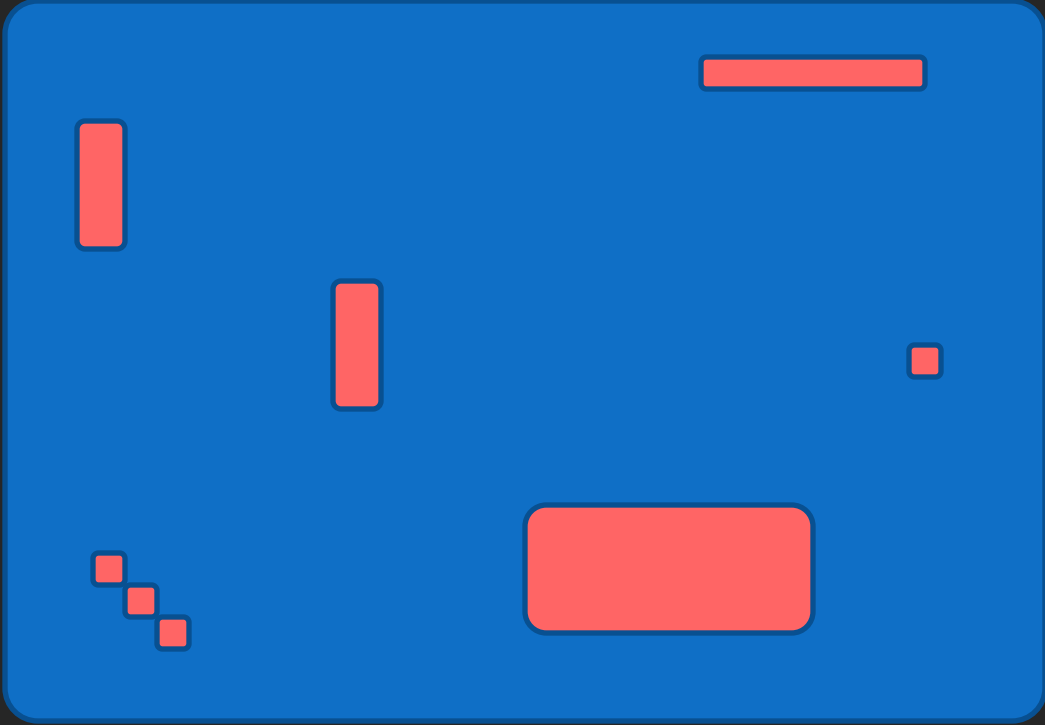




Imagine:  
Clean separation of state...



- Persist
- Store secrets
- PII



427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494

## Stateless Hardware (persistent state eliminated)

- Firmware infections prevented
- No places to store stolen secrets
- Reliable way to verify firmware
- Reliable way to *choose* firmware
- Boot multiple environments
- Share laptops with others



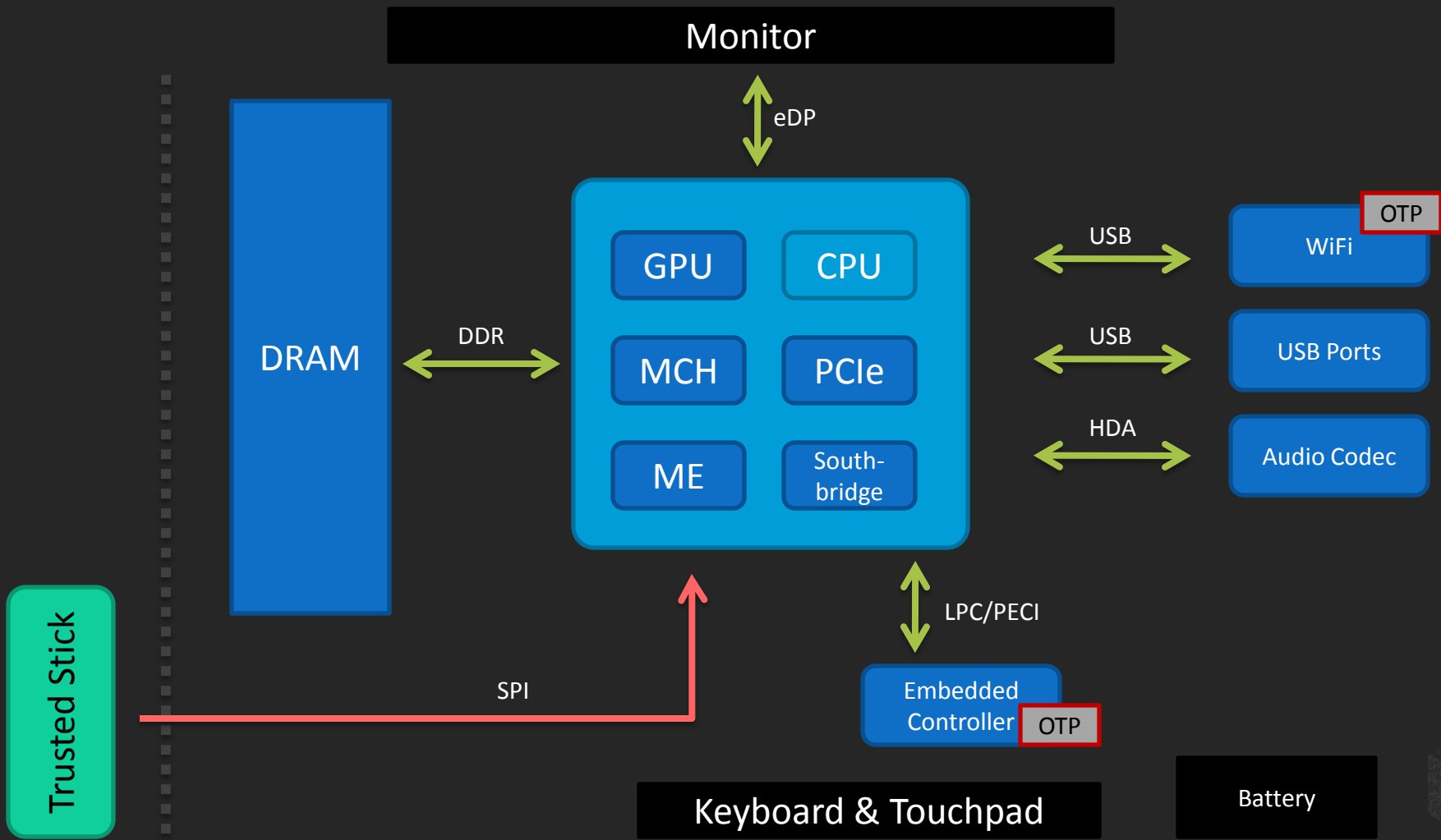
Trusted Stick



# Stateless Laptop

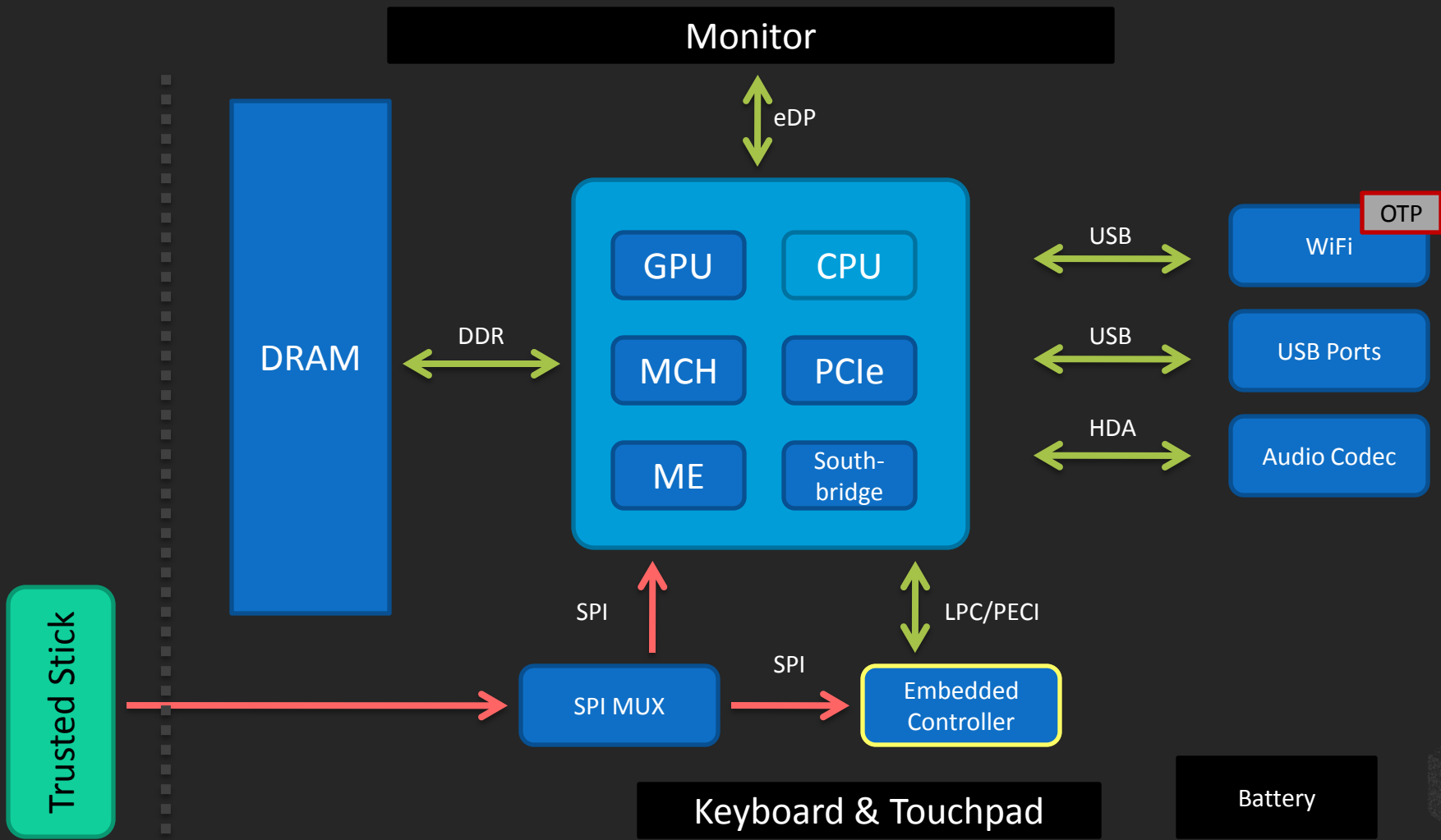
427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



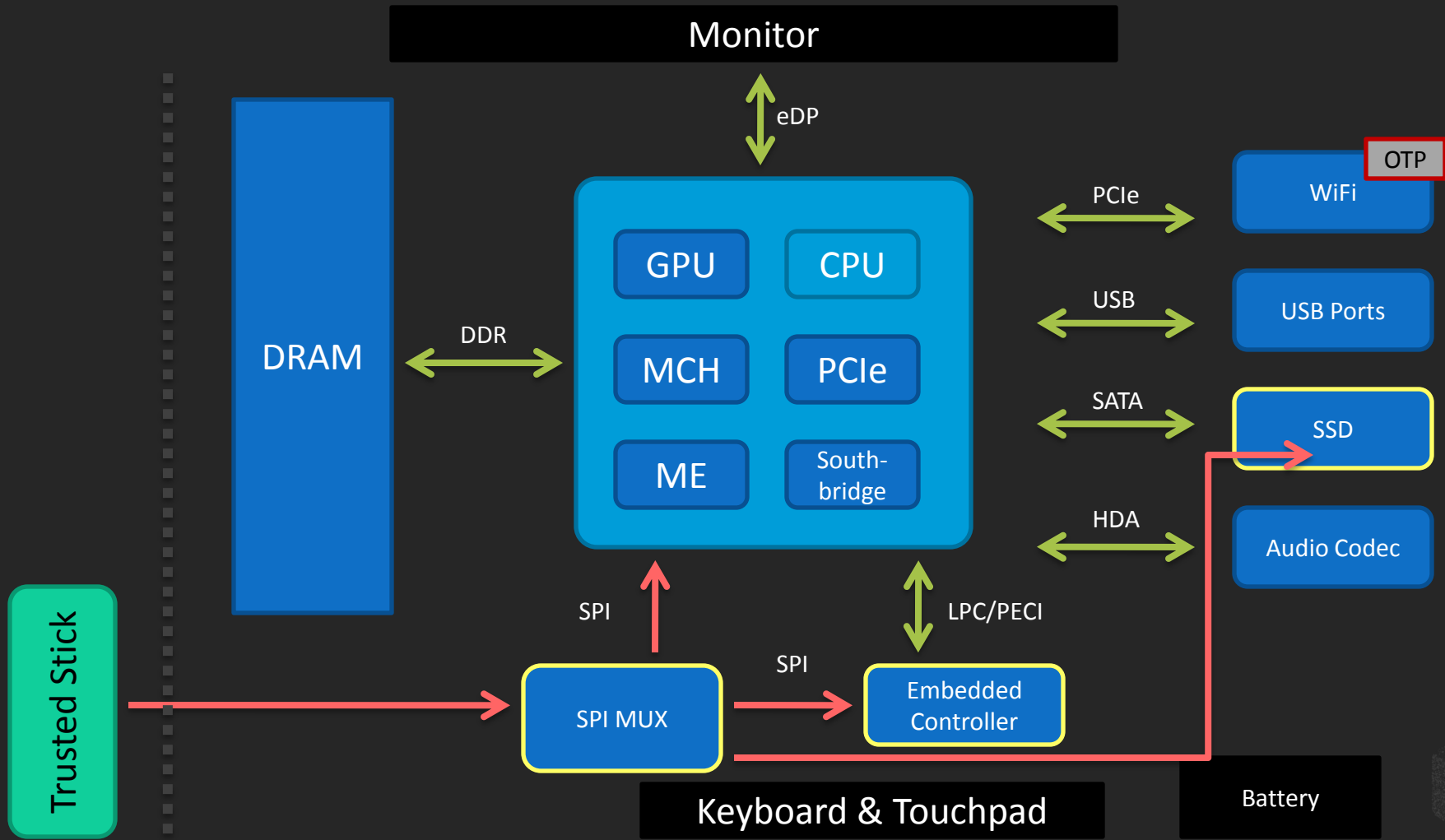


427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494









427F11FD 0FAA4B08 0123F01C DDF41A3E 36879494

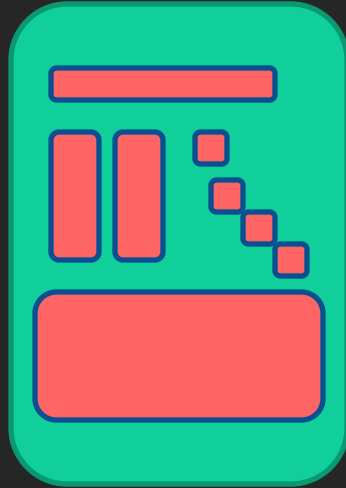


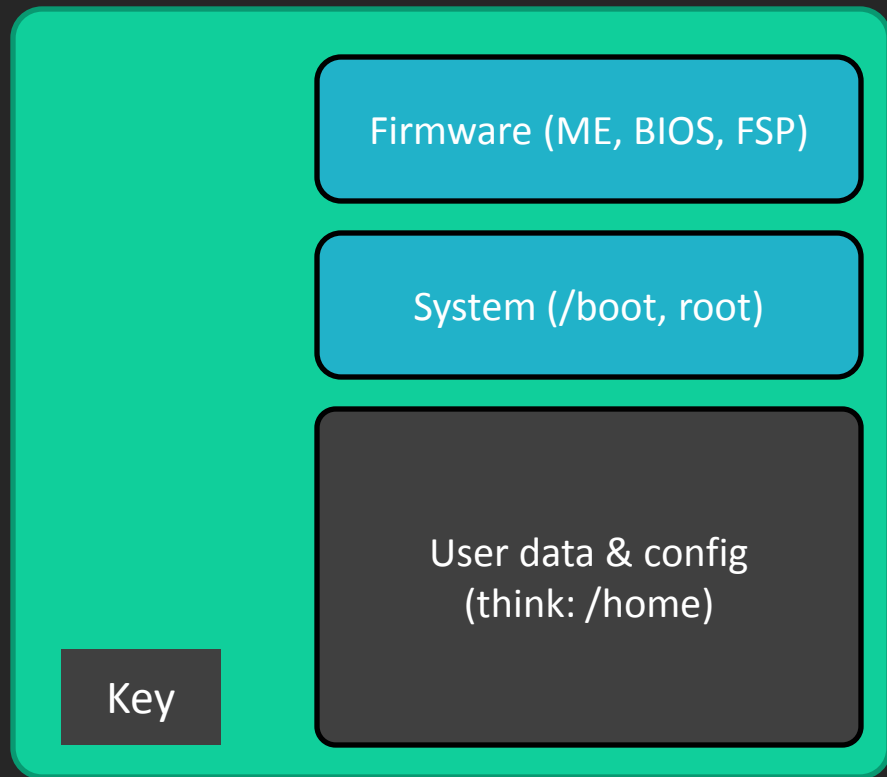
# But wait a sec... disk in a *stateless* laptop?!

It's a special disk and we will get back to this in a moment...

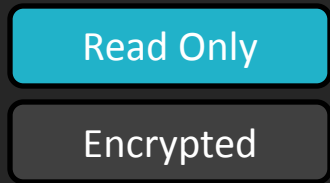


# Trusted Stick





Trusted Stick



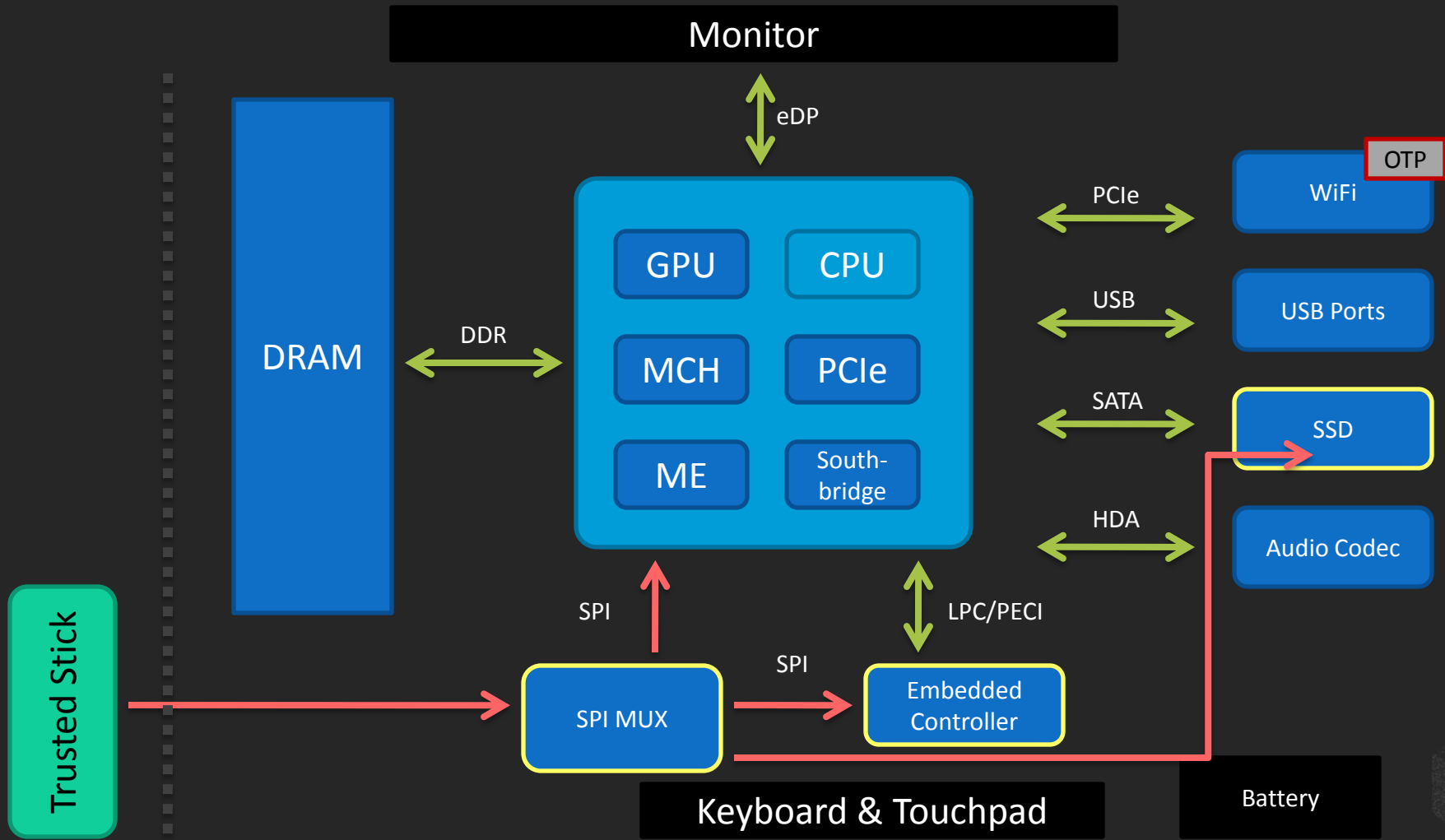
427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



# The (optional) internal disk

- R/O protection for system partitions
- Encryption for user partitions
- Trusted firmware!
- Open firmware, hardware!

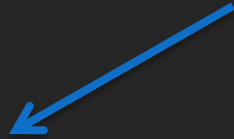




427F11FD 0FAA4B08 0123F01C DDF41A3E 36879494



# Leaks through networking



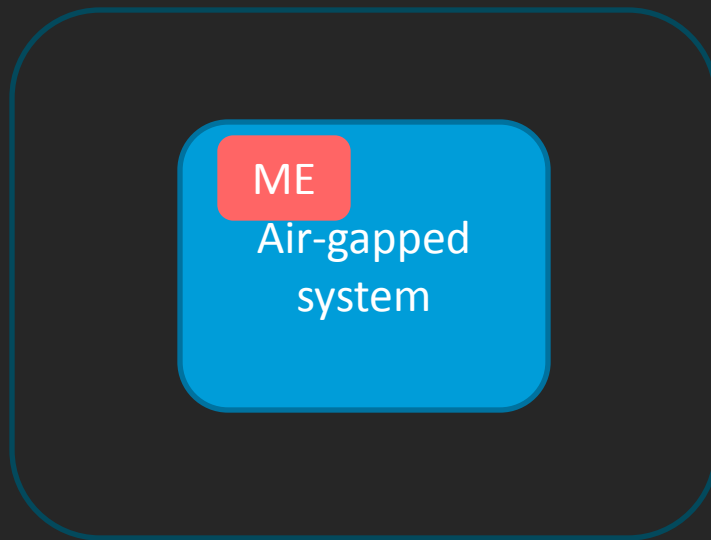
“Classic malware”



Sophisticated malware (e.g. ME)

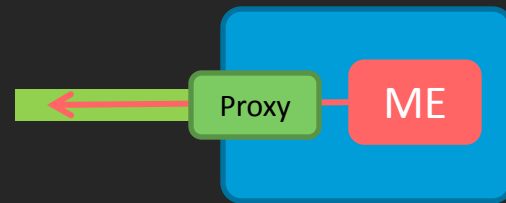
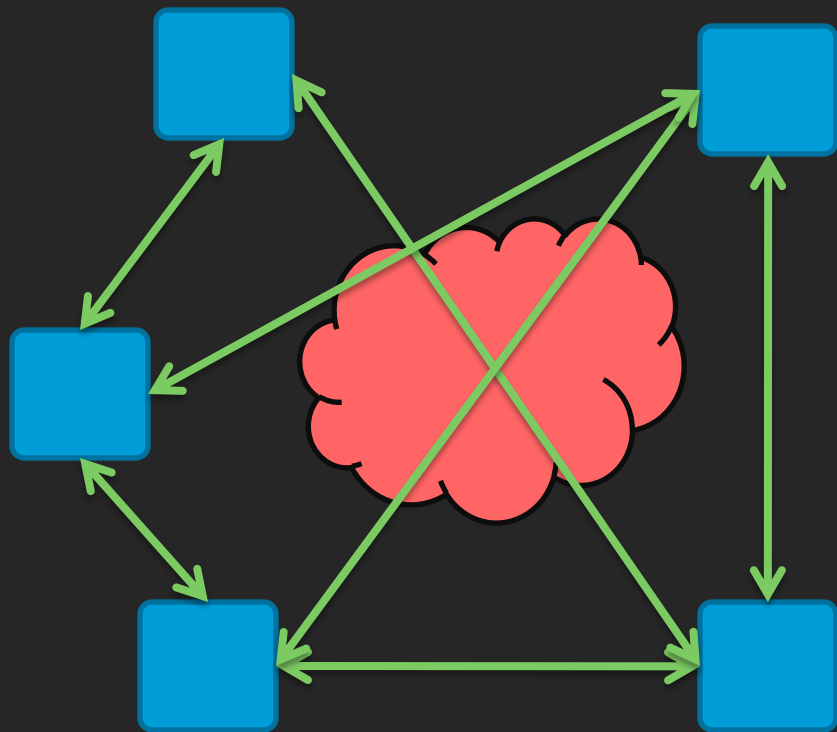


# Scenario #0

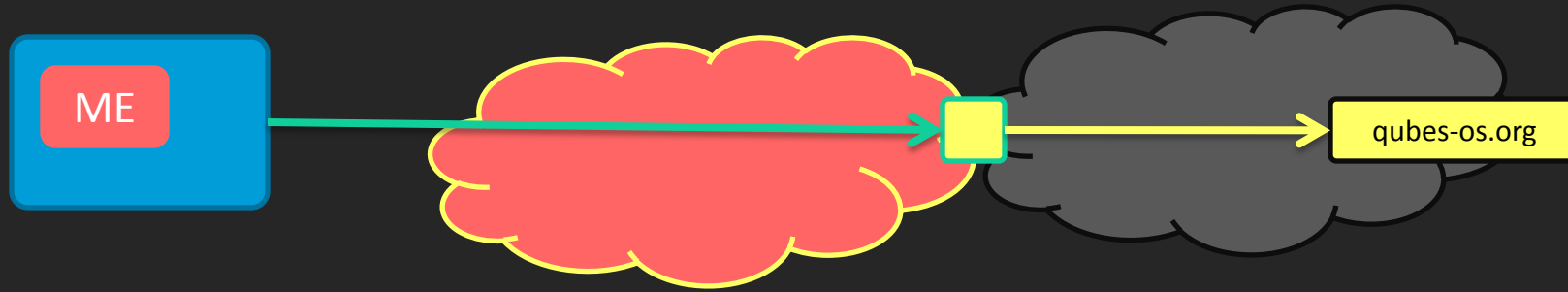




# Scenario #1 (Closed network of trusted nodes)



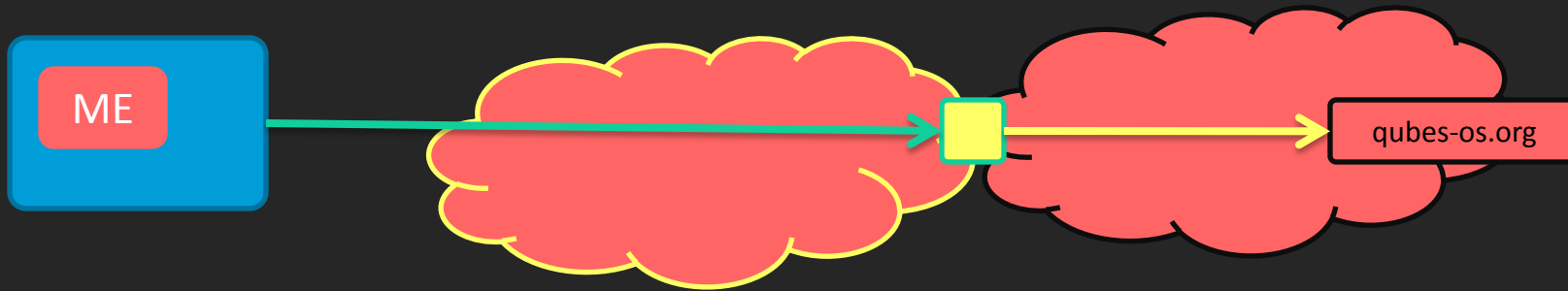
# Scenario #2 (Open network via Tor/VPN)



427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



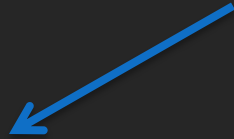
# Scenario #2 (Open network via Tor/VPN)



427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



# Leaks through networking



“Classic malware”



Sophisticated malware (e.g. ME)



# (Un)trusting the BIOS & host OS?

- Stick enforces R/O...
- Laptop provides no places for secrets...
- We don't care aboutt the BIOS!



# (Un)trusting the BIOS & host OS?

- Stick enforces R/O...
- Laptop provides no places for secrets...
- We don't care aboutt the BIOS!
- Yet BIOS might provide privileges escalation
- And we don't want "classic malware"!



# So, we want BIOS without backdoors!

- ME injecting classic malware?
- FSP?

Yes, thankfully we have coreboot! :)



# Evil Maid Attacks

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494

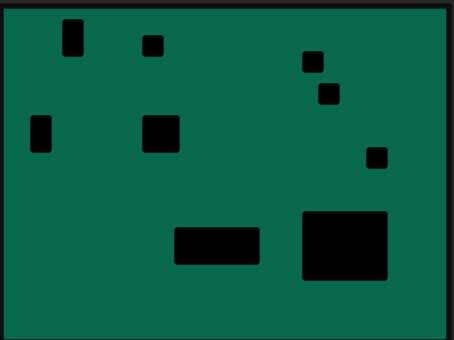




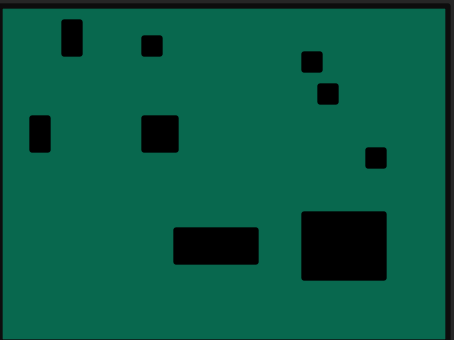
How do you know you got a true *stateless* laptop?

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494





?



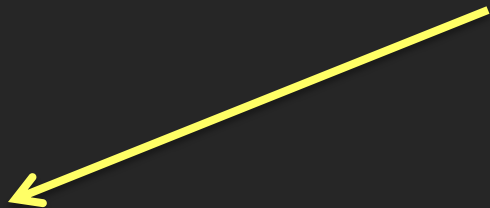


# Other architectures?

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



# Alternatives?



## ARM?

- The “ARM processor”?
- (Too much) diversity
- Trust Zone
- Backdoors still



# Alternatives?



## ARM?

- The “ARM processor”?
- (Too much) diversity
- Trust Zone
- Backdoors still

## Open source CPUs?

- Performance?
- Security technologies?
- Availability?



# Alternatives?



```
graph TD; A[Alternatives?] -- yellow arrow --> B[ARM?]; A -- cyan arrow --> C[Open source CPUs?]
```

## ARM?

- The “ARM processor”?
- (Too much) diversity
- Trust Zone
- Backdoors still

## Open source CPUs?

- Performance?
- Security technologies?
- Availability?

Evil Maid Attacks still apply, BTW!

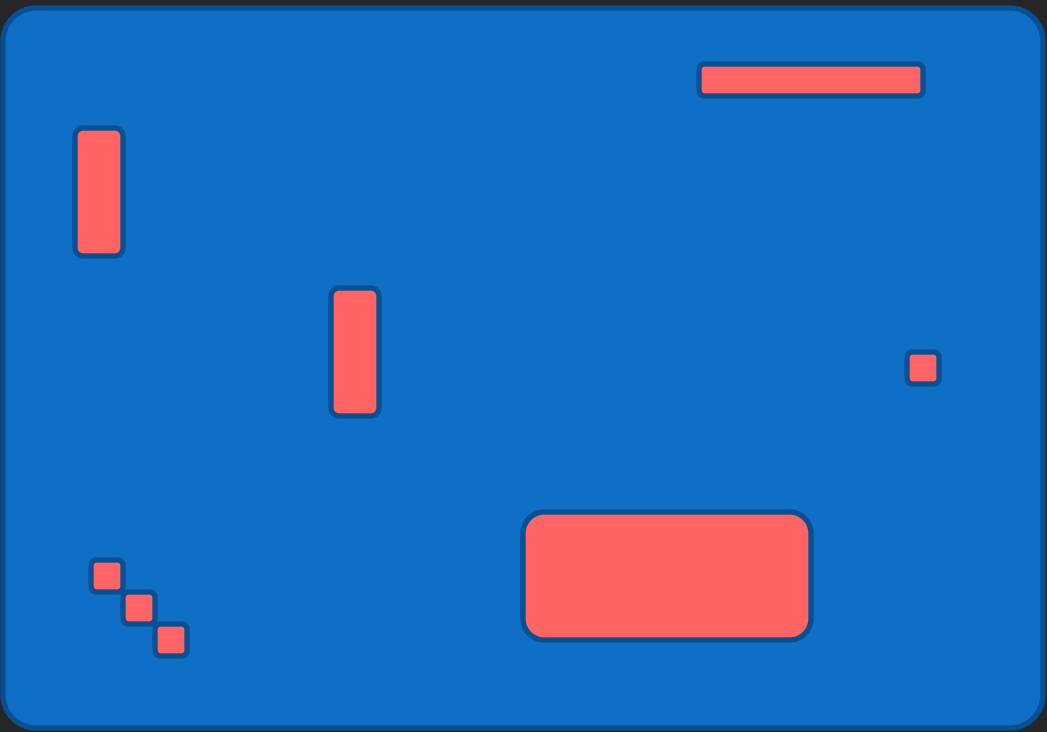


Clean state separation still makes sense  
#IMHO



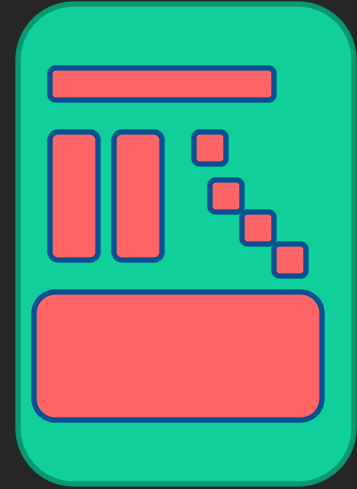


427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



Stateless Hardware  
(largely untrusted)

- Firmware infections prevented
- No places to store stolen secrets
- Reliable way to verify firmware
- Reliable way to *choose* firmware
- Boot multiple environments
- Share laptops with others



Trusted Stick



# Market forces?



Market forces  
vs.  
Human & Civil rights?



# Legislation to the rescue!





Our world depends on computers...

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494



# Thank you!

Name	Size
 x86_harmful.pdf	343.3 kB
 state_harmful.pdf	278.5 kB

427F11FD 0FAA4B08 0123F01C DDFa1A3E 36879494

