Formally Verified Isolation of DMA

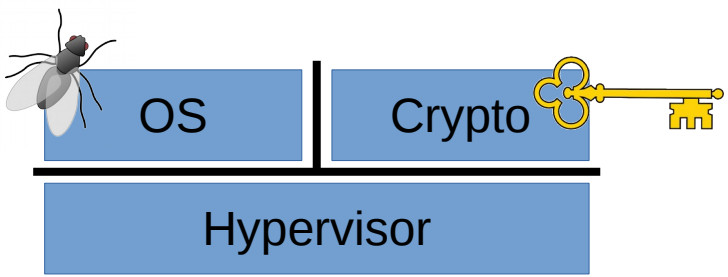
Jonas Haglund, Roberto Guanciale

KTH Royal Institute of Technology,
Stockholm, Sweden

FMCAD, 19 October 2022

Critical Memory Isolation

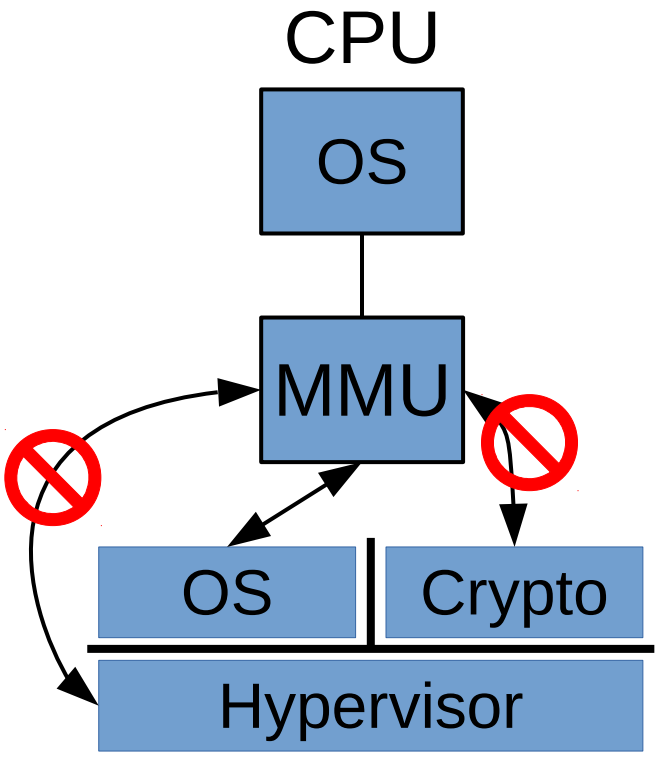
CertiKOS



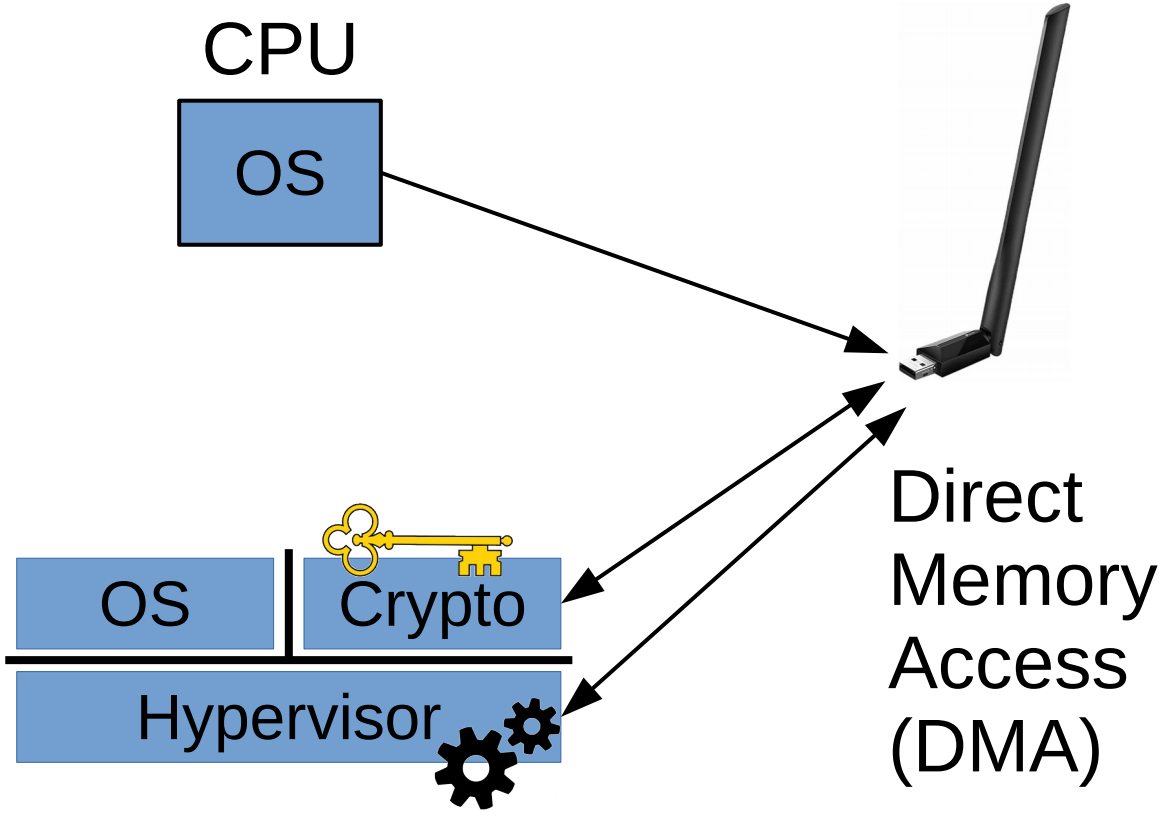
PROSPER

$$\frac{A \qquad B}{A \wedge B}$$

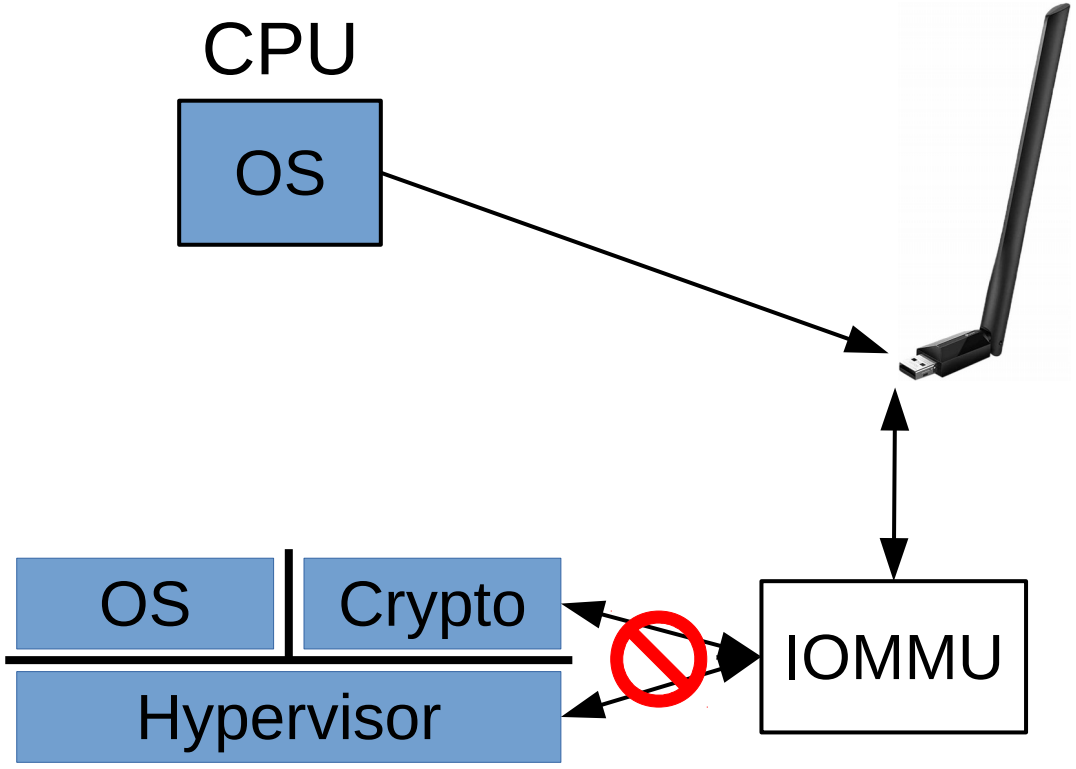
Memory Isolation of Software



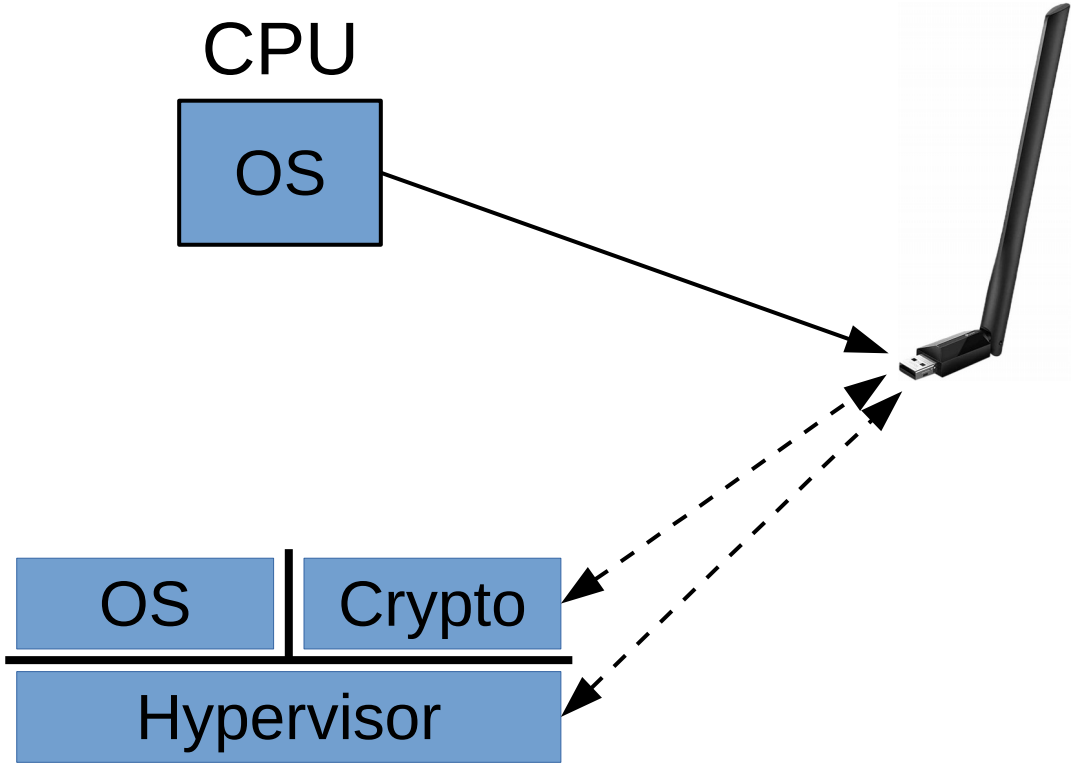
Peripherals Breaking Memory Isolation



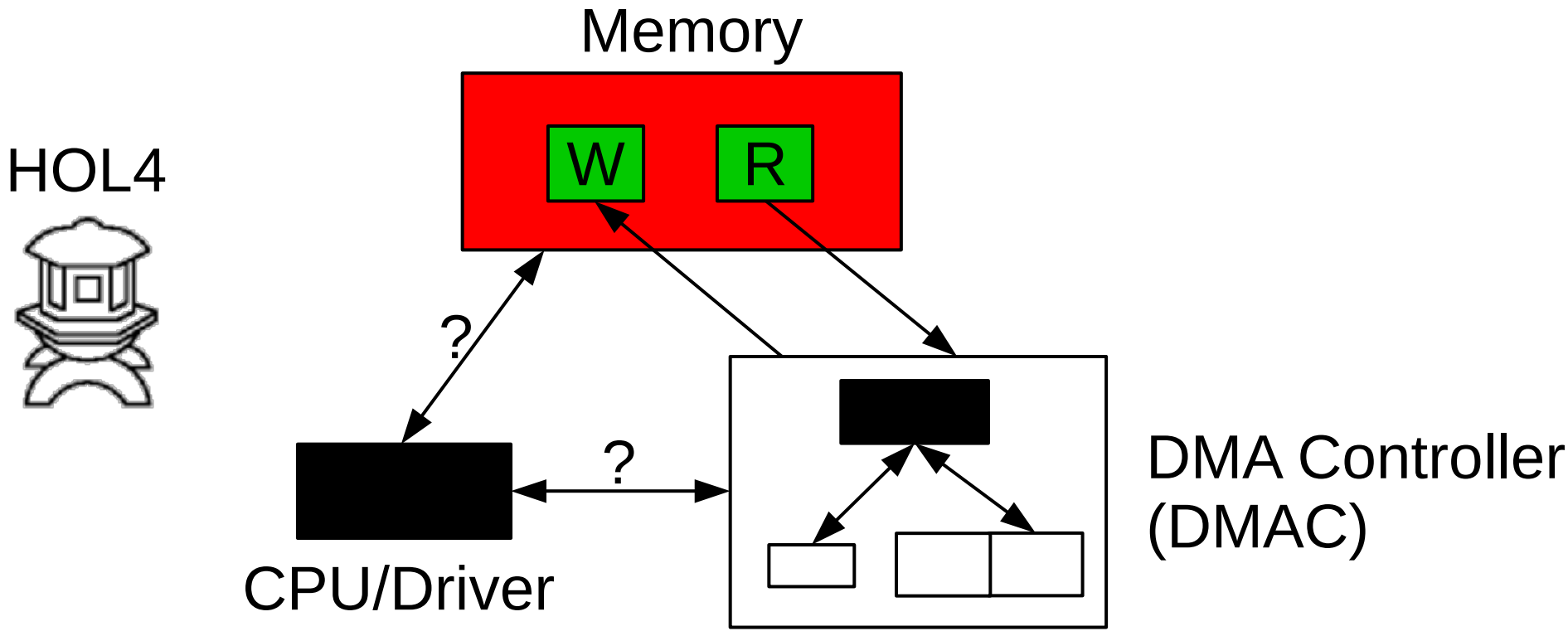
Memory Isolation with IOMMU



Memory Isolation without IOMMU

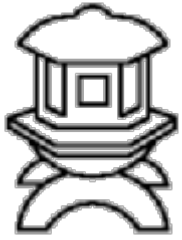


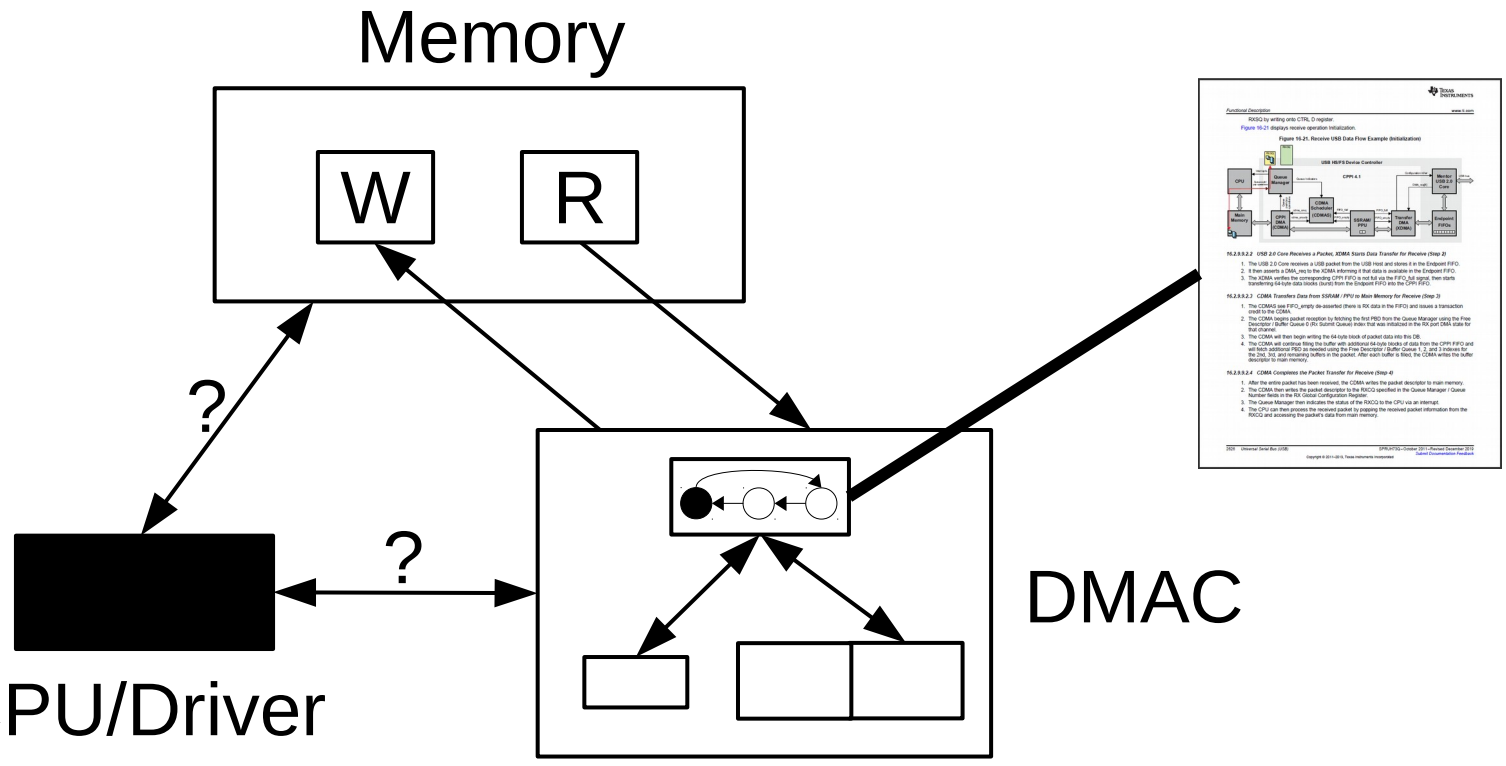
Contribution: Formally Verified DMAC Isolation



Contribution: Formally Verified DMAC Isolation

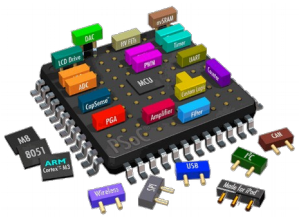
HOL4

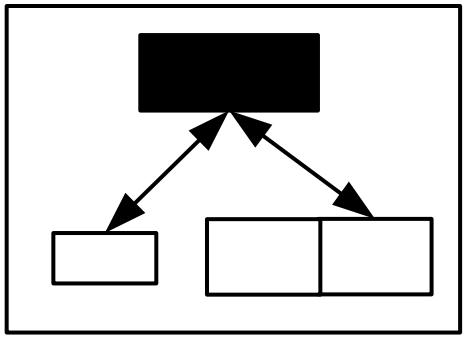




Generality of Framework

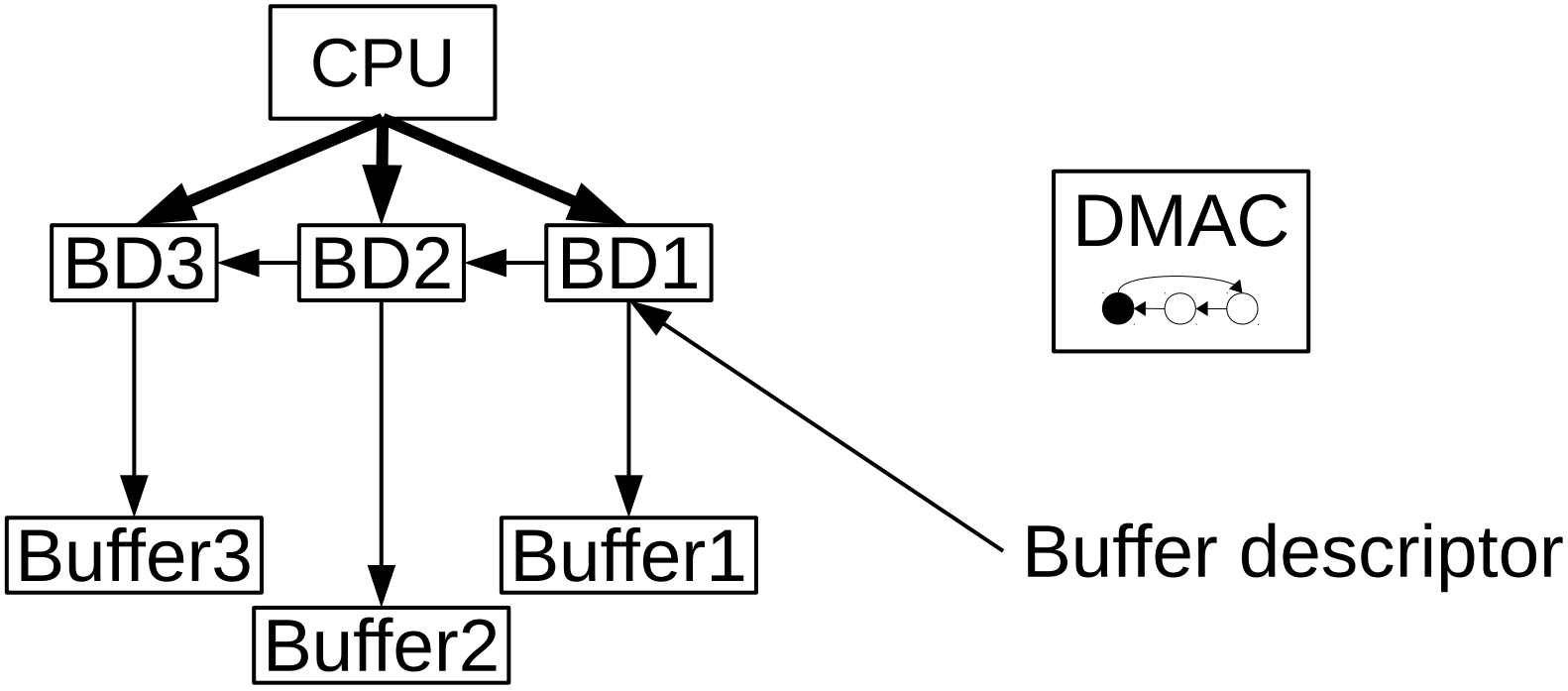
>20 DMACs studied



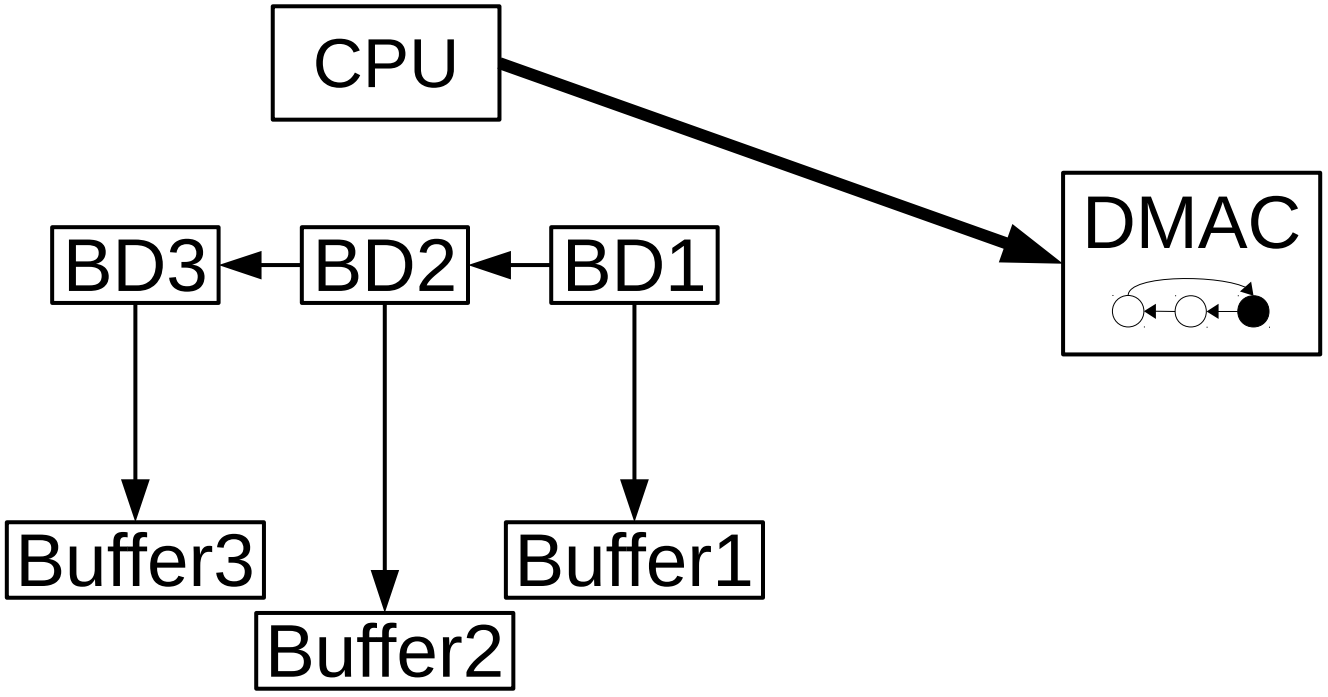


DMAC

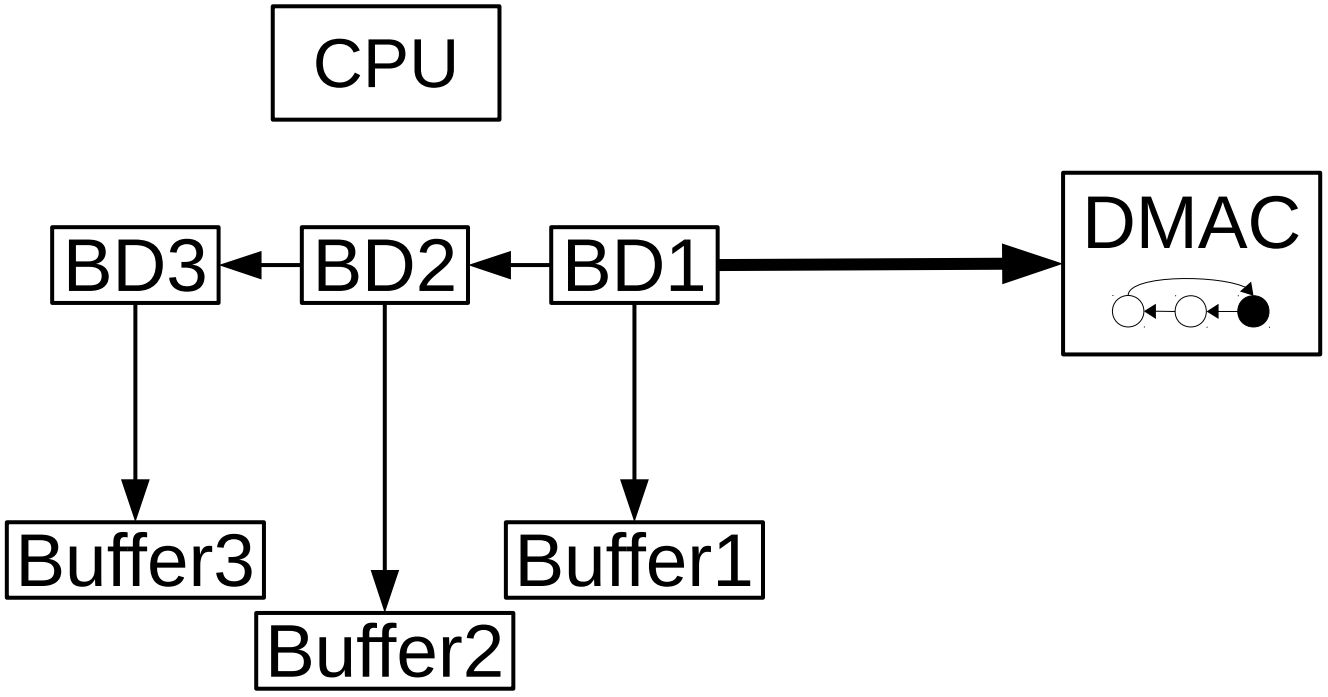
DMA: CPU Initializing BDs



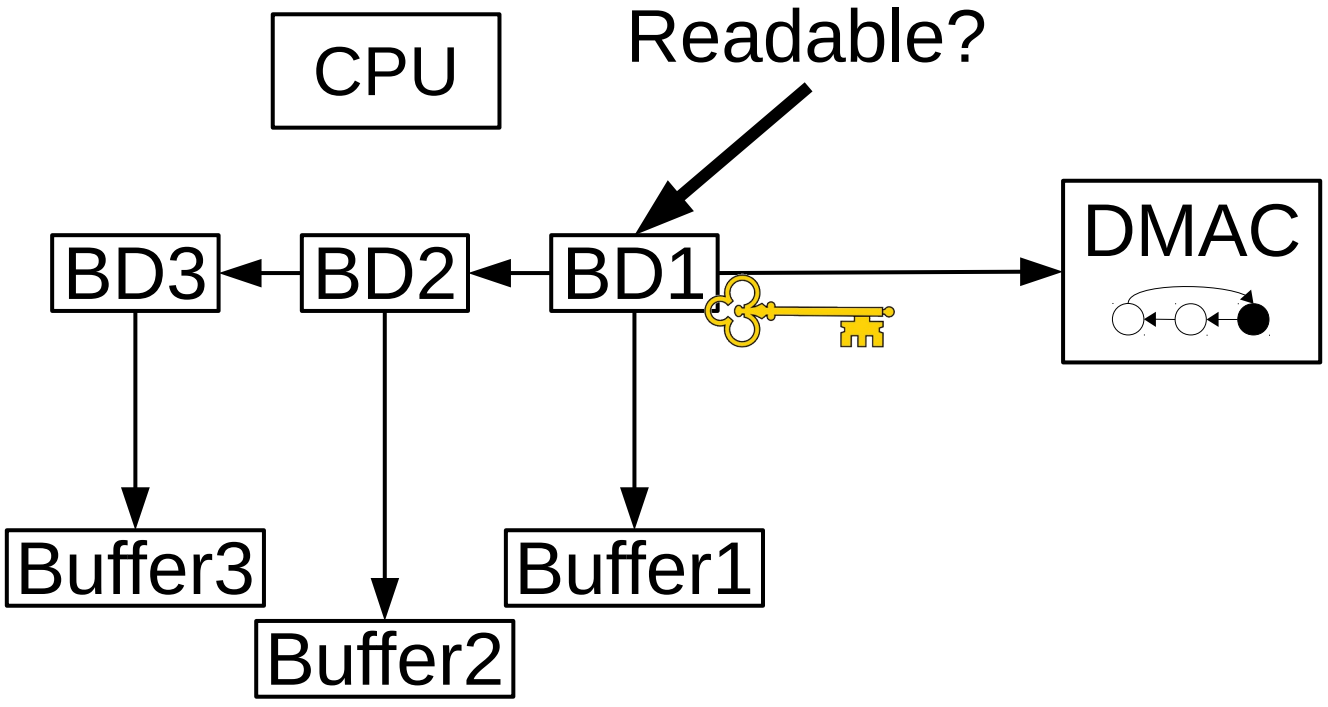
DMA: CPU Activating DMAC



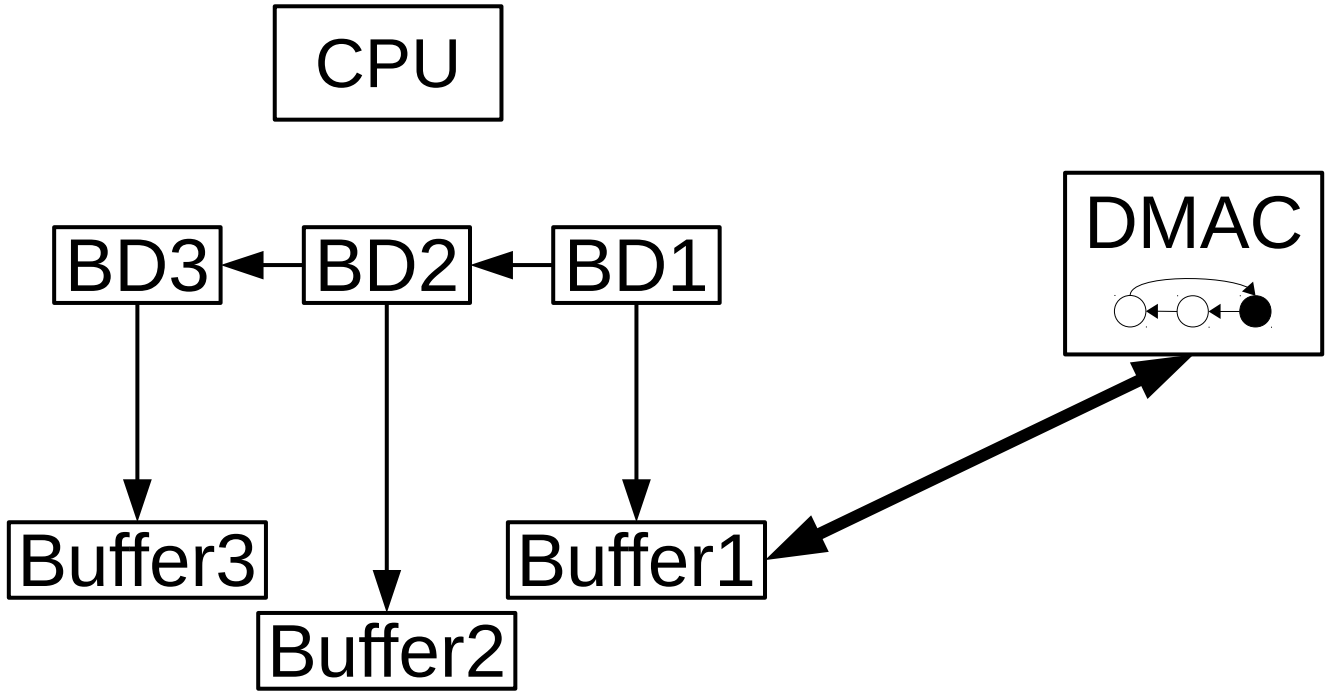
DMA: DMAC Fetching BD



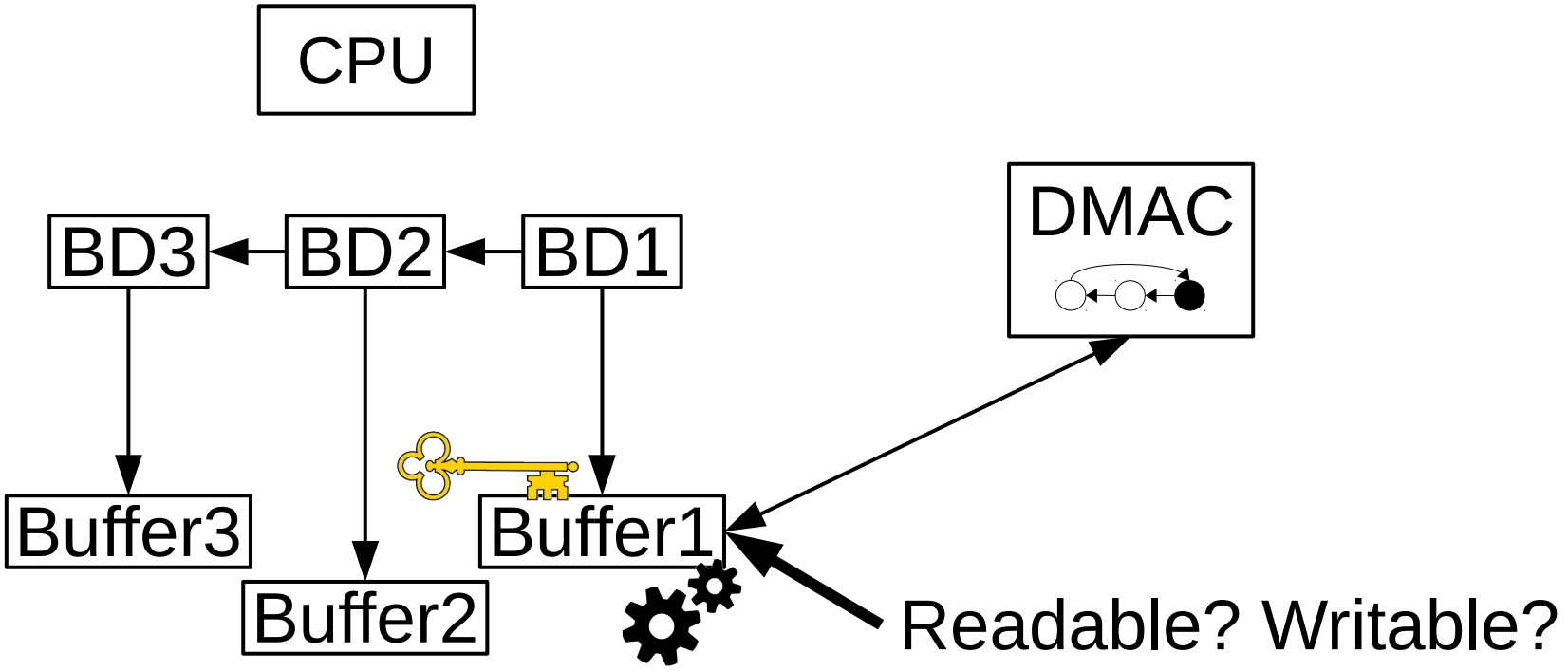
DMA: DMAC Fetching BD



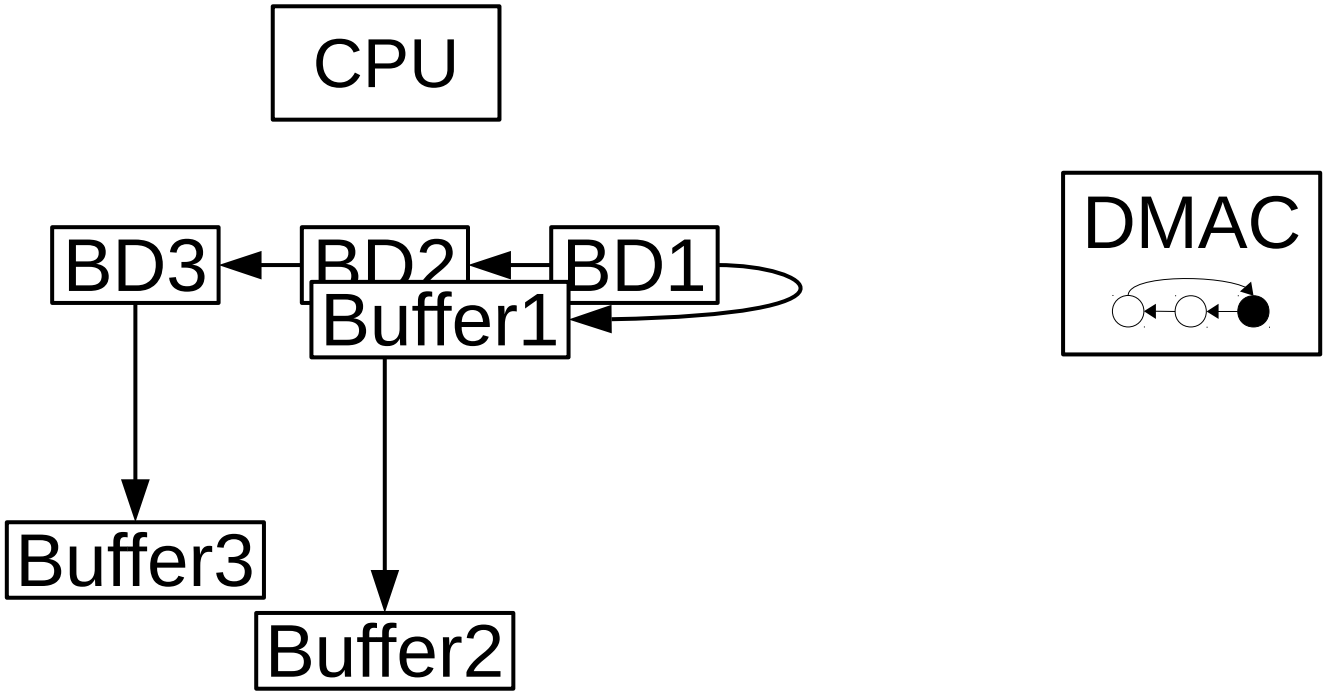
DMA: DMAC Processing BD



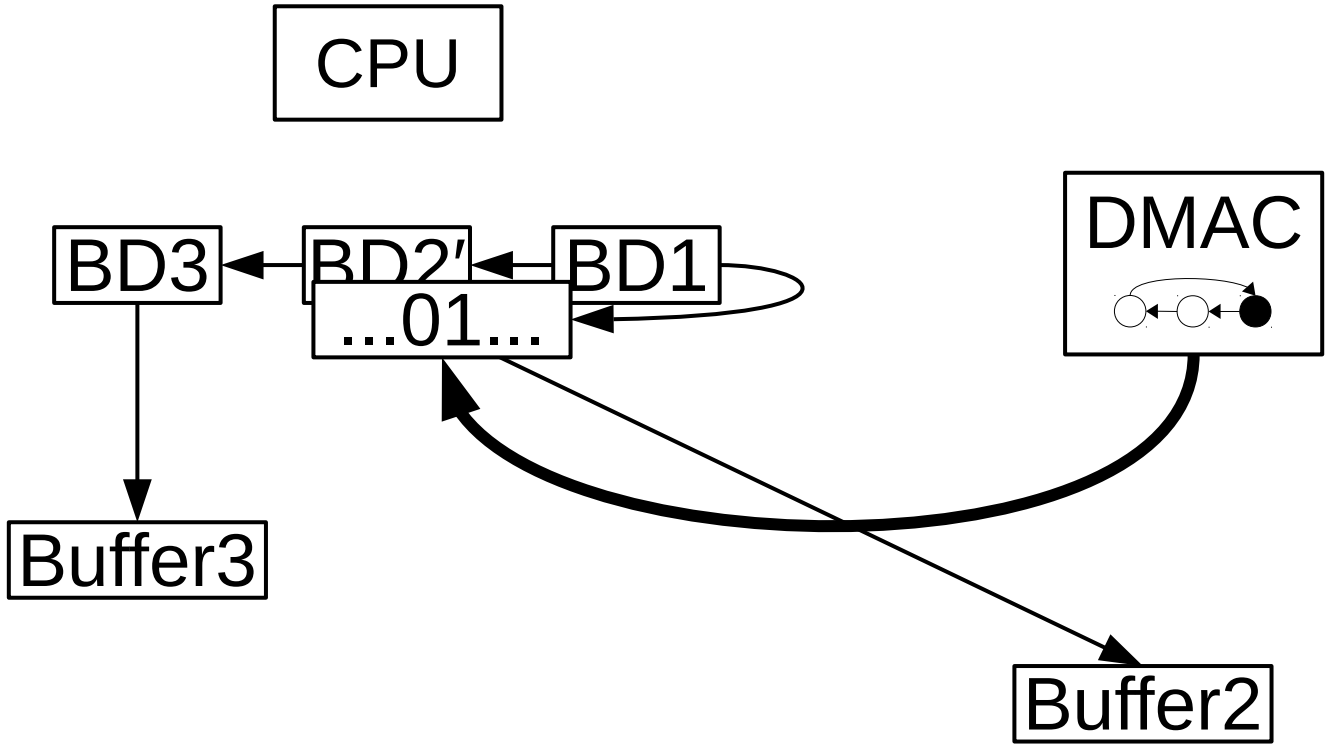
DMA: DMAC Processing BD



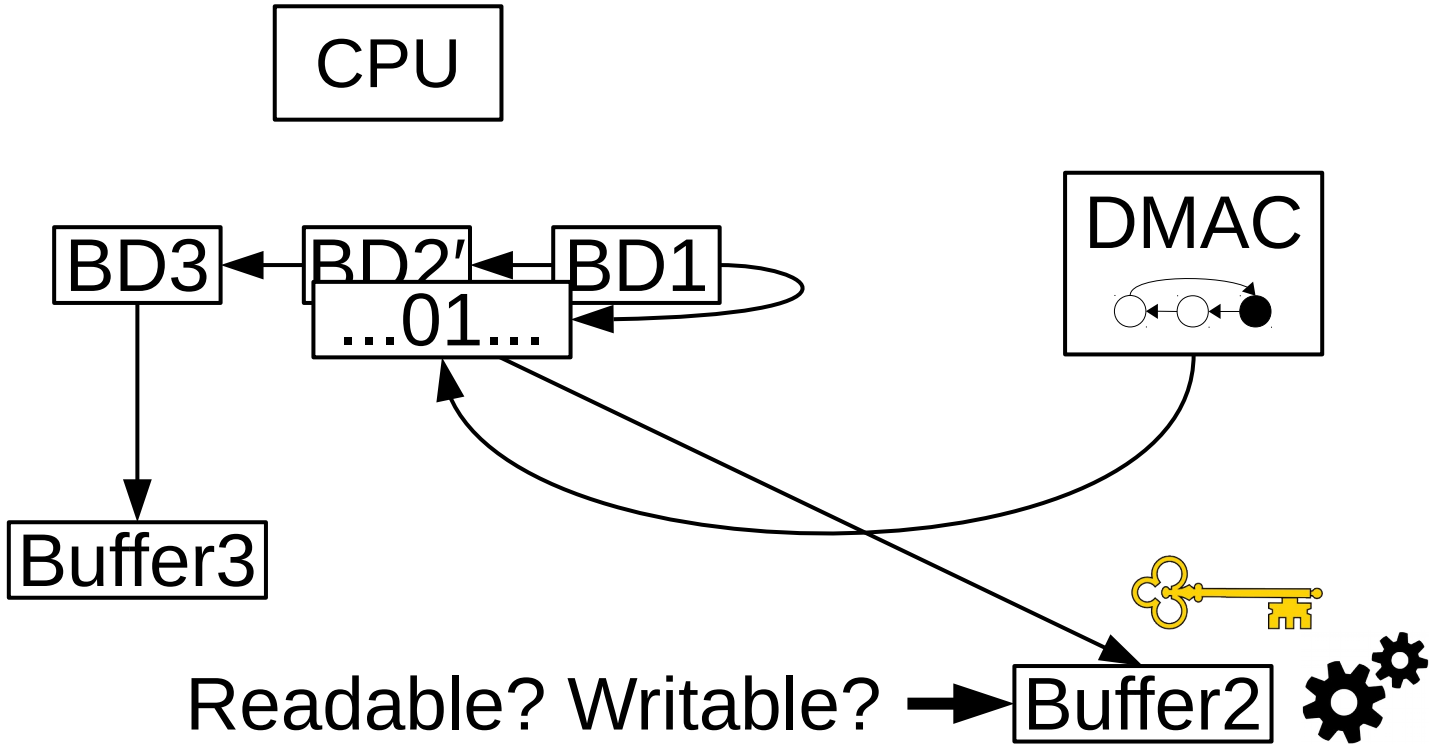
DMA: DMAC Processing BD



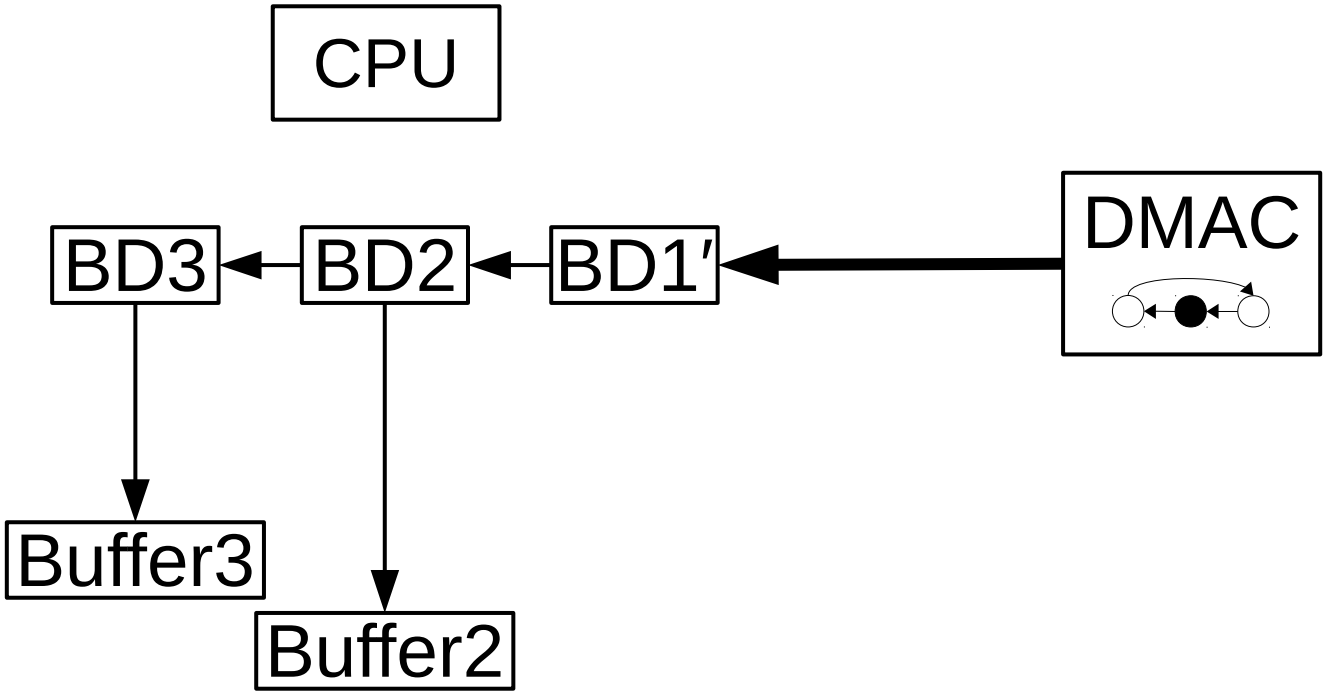
DMA: DMAC Processing BD



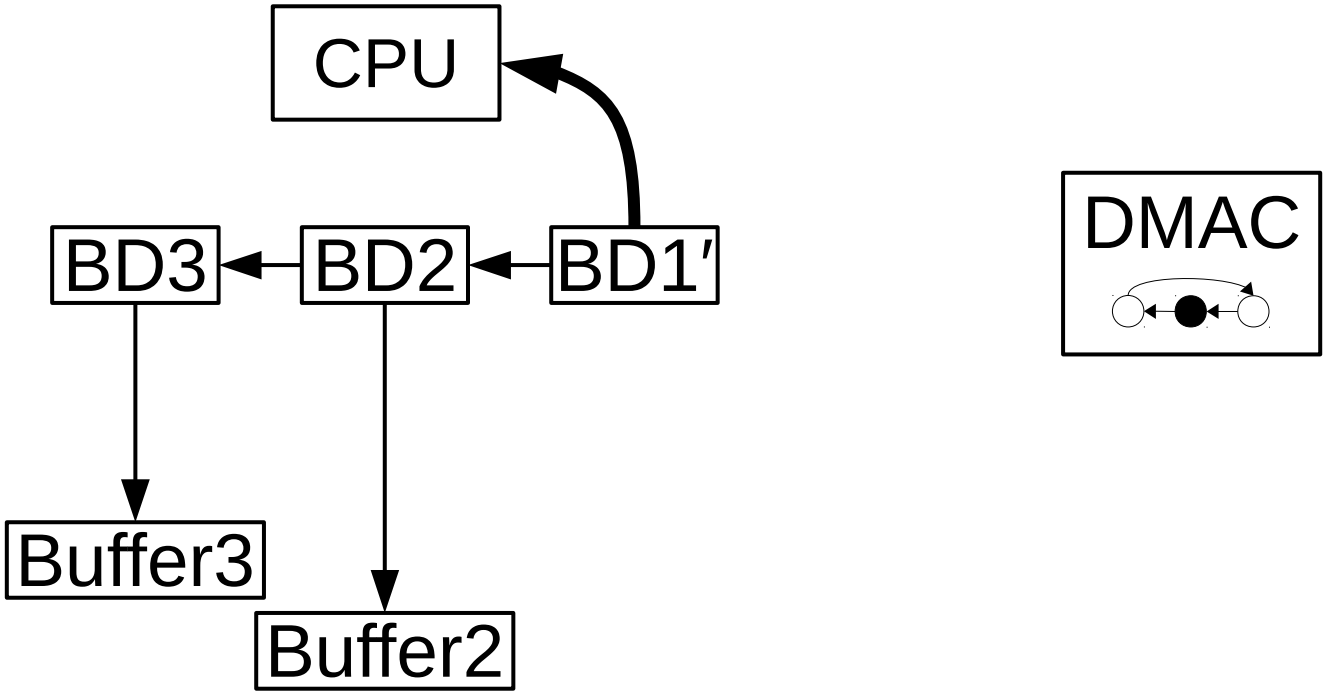
DMA: DMAC Processing BD



DMA: DMAC Writing Back BD



DMA: DMAC Writing Back BD



Labeled Transition System

$cpu_1 \xrightarrow{\tau} cpu_2$ $cpu_1 \xrightarrow{read(as,\ bs)} cpu_2$ $cpu_1 \xrightarrow{write(as,\ bs)} cpu_2$

Labeled Transition System

$$cpu_1 \xrightarrow{\tau} cpu_2 \qquad cpu_1 \xrightarrow{read(as,\ bs)} cpu_2 \qquad cpu_1 \xrightarrow{write(as,\ bs)} cpu_2$$

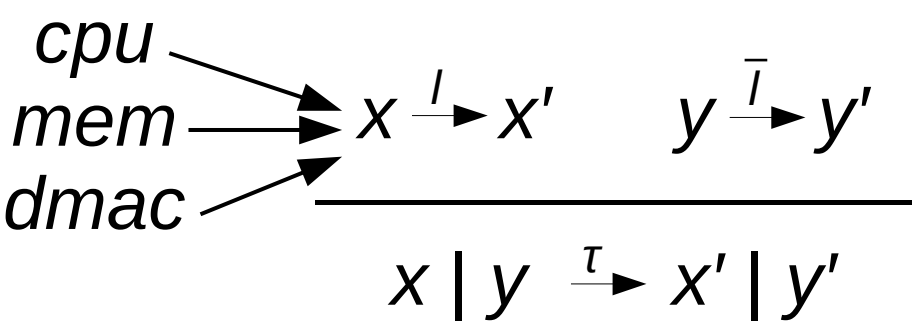
$$mem \xrightarrow{\overline{read(as,\ mem(as))}} mem \qquad mem \xrightarrow{\overline{write(as,\ bs)}} mem[as \mapsto bs]$$

Labeled Transition System

$$cpu_1 \xrightarrow{\tau} cpu_2 \qquad cpu_1 \xrightarrow{read(as,\, bs)} cpu_2 \qquad cpu_1 \xrightarrow{write(as,\, bs)} cpu_2$$

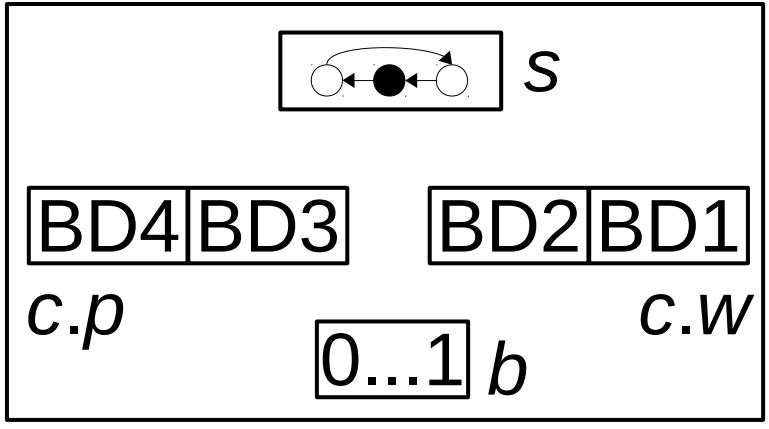
$$mem \xrightarrow{\overline{read(as,\, mem(as))}} mem \qquad mem \xrightarrow{\overline{write(as,\, bs)}} mem[as \mapsto bs]$$

$$\frac{x \xrightarrow{\tau} x'}{x \mid y \xrightarrow{\tau} x' \mid y}$$



DMAC State (*s*, *b*, *c*)

- *s*: DMAC specific state
- *b*: Memory requests/replies
- *c*: channel of BDs: *c*.*p*, *c*.*w*



DMAC Transitions: Processing BDs

DMAC specific

Completion flag

$$\boldsymbol{p}(s_1, \boldsymbol{rp}, bd) = (s_2, \boldsymbol{rq}, \mathrm{T}) \qquad c.p = bd::bds$$

$$(s_1, b + \boldsymbol{rp}, c) \xrightarrow{\tau} (s_2, b + \boldsymbol{rq}, c := \{p := bds,\ w := c.w +\!+ bd\})$$

Reply

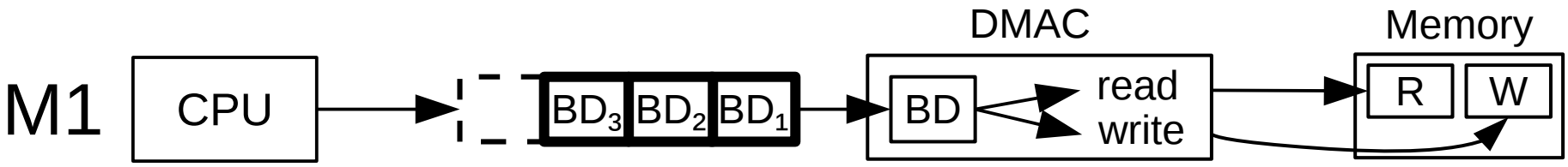
New request

DMAC Transitions: DMA

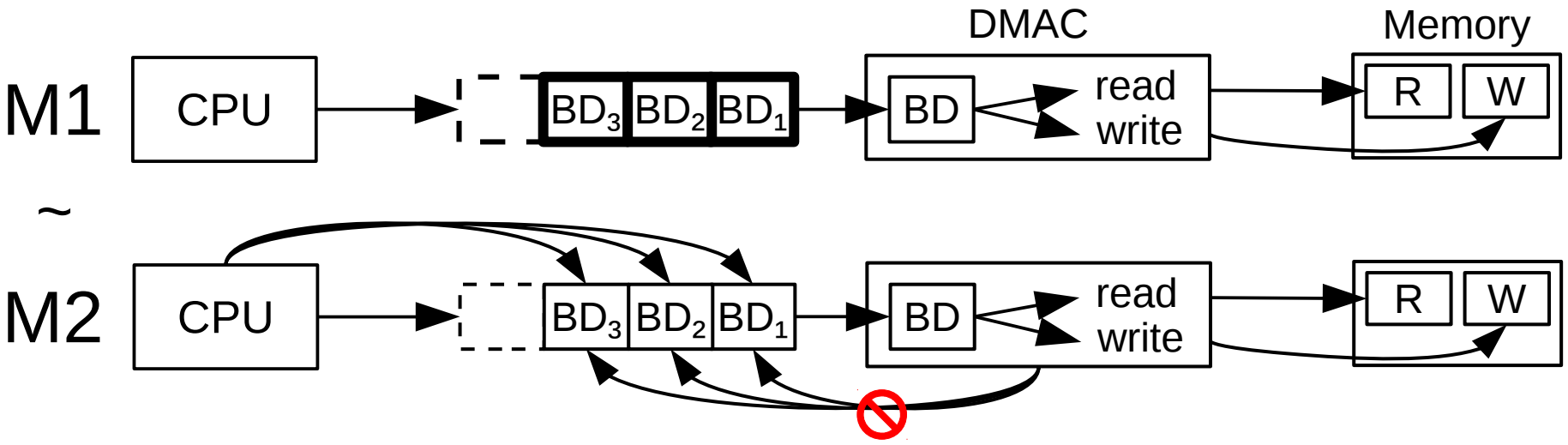
$$(s,\ b + \boldsymbol{rr_t(as)},\ c) \xrightarrow{read(as,\ bs)} (s,\ b + \boldsymbol{rp_t(bs)},\ c)$$

$$(s,\ b + \boldsymbol{wr(as,\ bs)},\ c) \xrightarrow{write(as,\ bs)} (s,\ b,\ c)$$

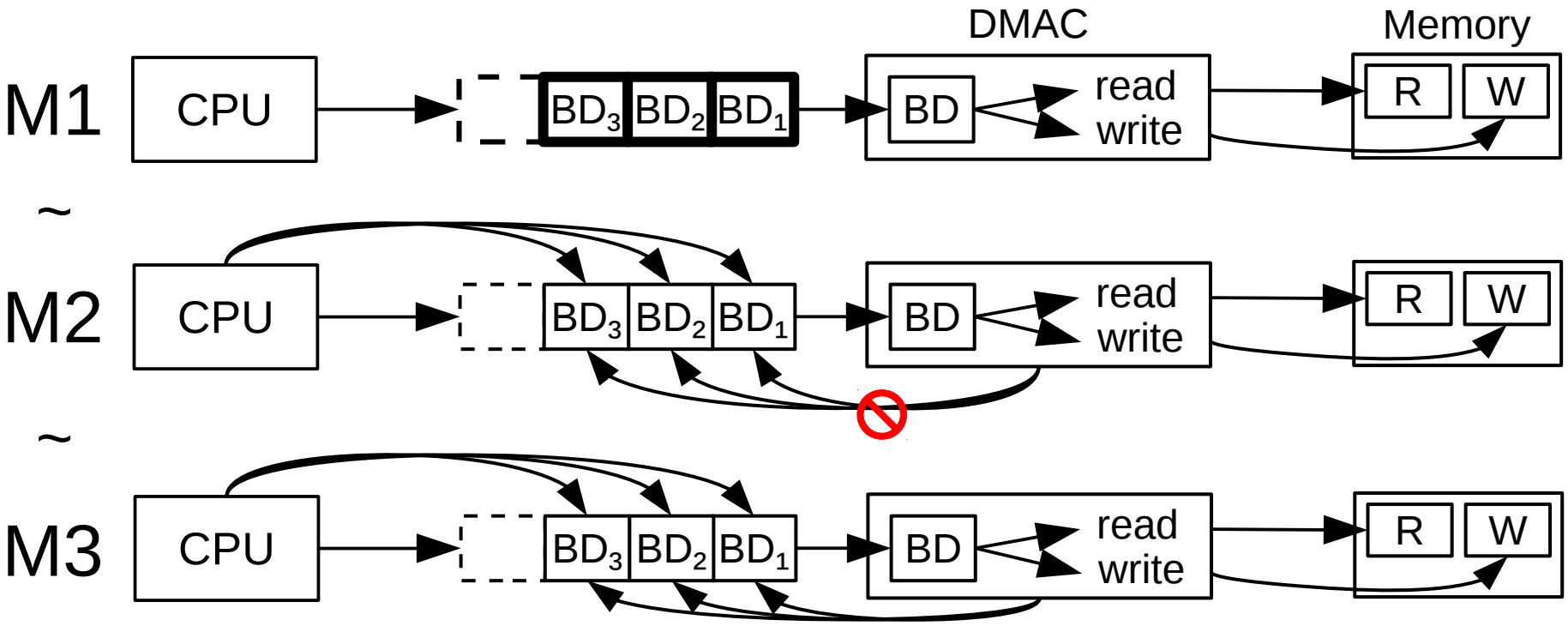
Framework: Isolated DMAC



Framework: Isolated DMAC



Framework: Isolated DMAC



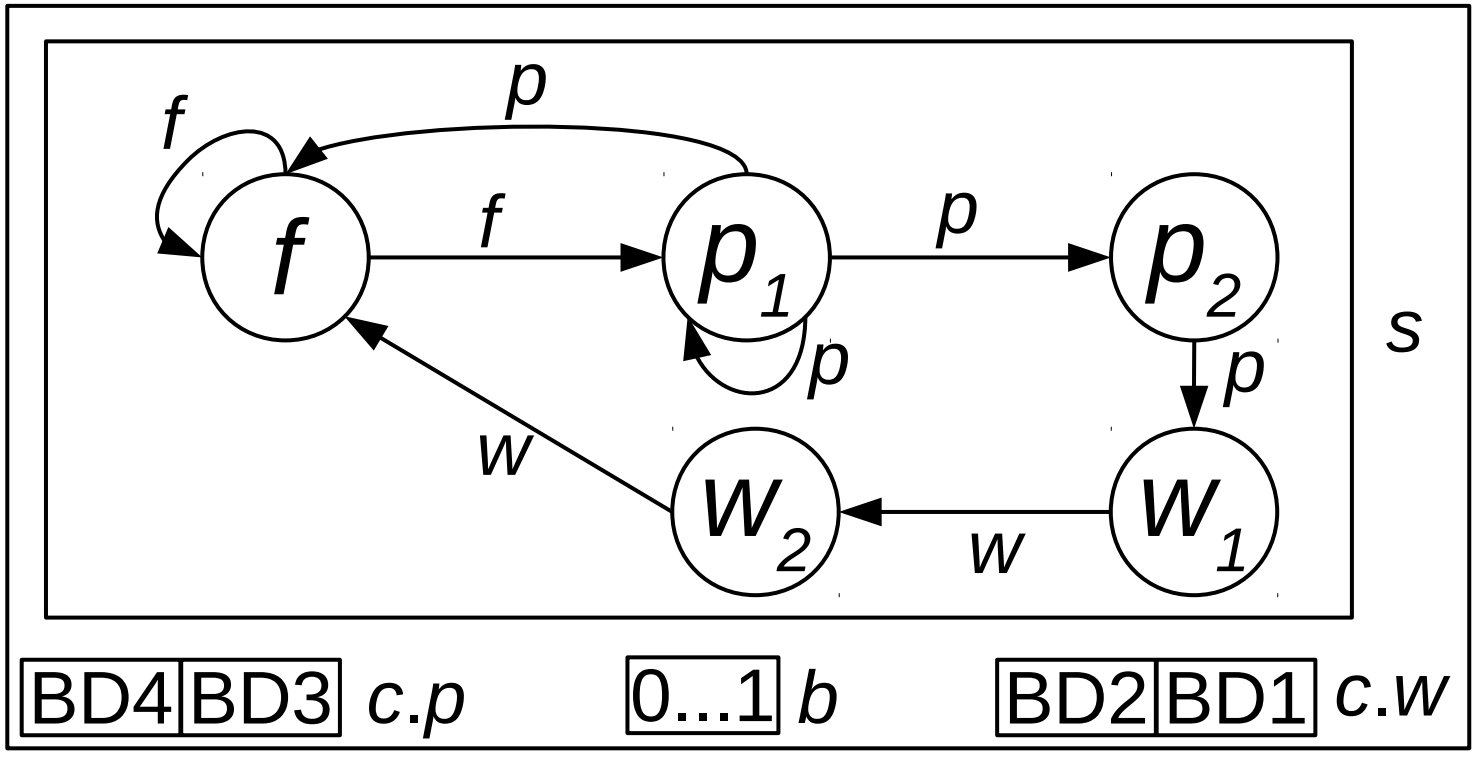
Formally Verified Memory Isolation

$$(cpu, dmac, mem) \xrightarrow{read(as,\ bs)} (cpu', dmac', mem') \Rightarrow as \subseteq R$$

$$(cpu, dmac, mem) \xrightarrow{write(as,\ bs)} (cpu', dmac', mem') \Rightarrow as \subseteq W$$

DMAC Proof Obligations

Instantiation: USB DMAC

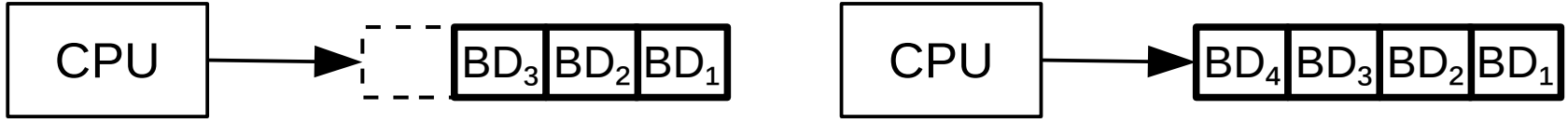


Evaluation: Formalization in HOL4

	NIC without framework	USB with framework
Verification Time	9 months	½ month
LoC	55000	2000

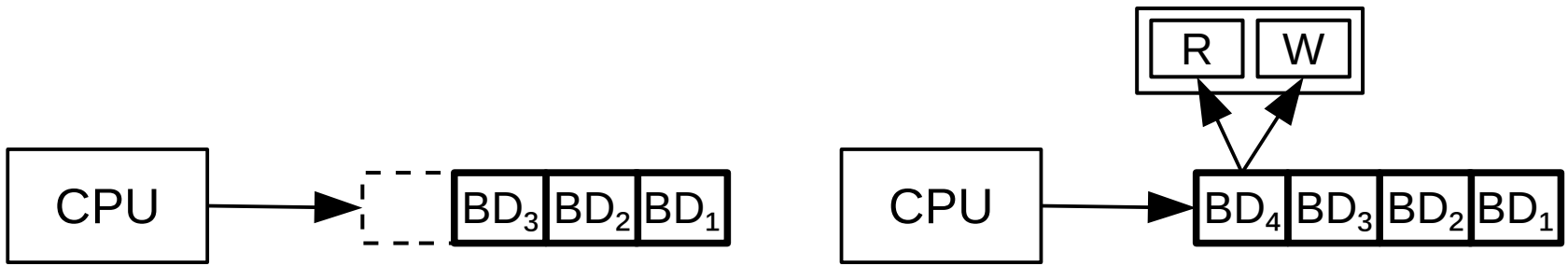
Concluding Remarks: DMAC Driver

- Driver modifies BD queues by only:
  - appending BDs



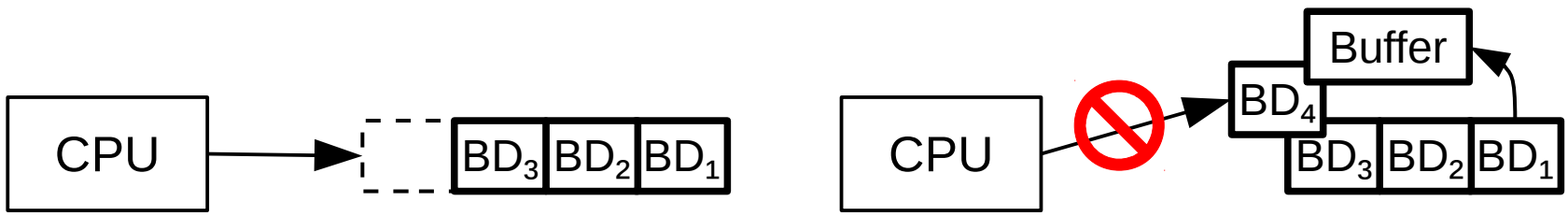
Concluding Remarks: DMAC Driver

- Driver modifies BD queues by only:
  - appending BDs
  - appending BDs that address R and W



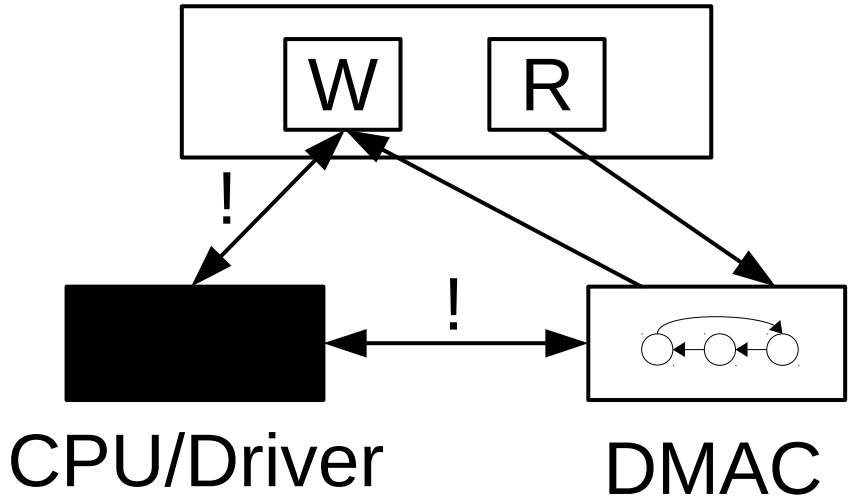
Concluding Remarks: DMAC Driver

- Driver modifies BD queues by only:
  - appending BDs
  - appending BDs that address R and W
  - appending BDs not overlapping each other or buffers



Concluding Remarks: DMAC Driver

- Driver modifies BD queues by only...
- ..., then the DMAC is isolated



Thank you!

Questions?