Hardware-assisted memory tracing on new SoCs embedding FPGA fabrics

Letitia Li, Guillaume Duc, Renaud Pacalet

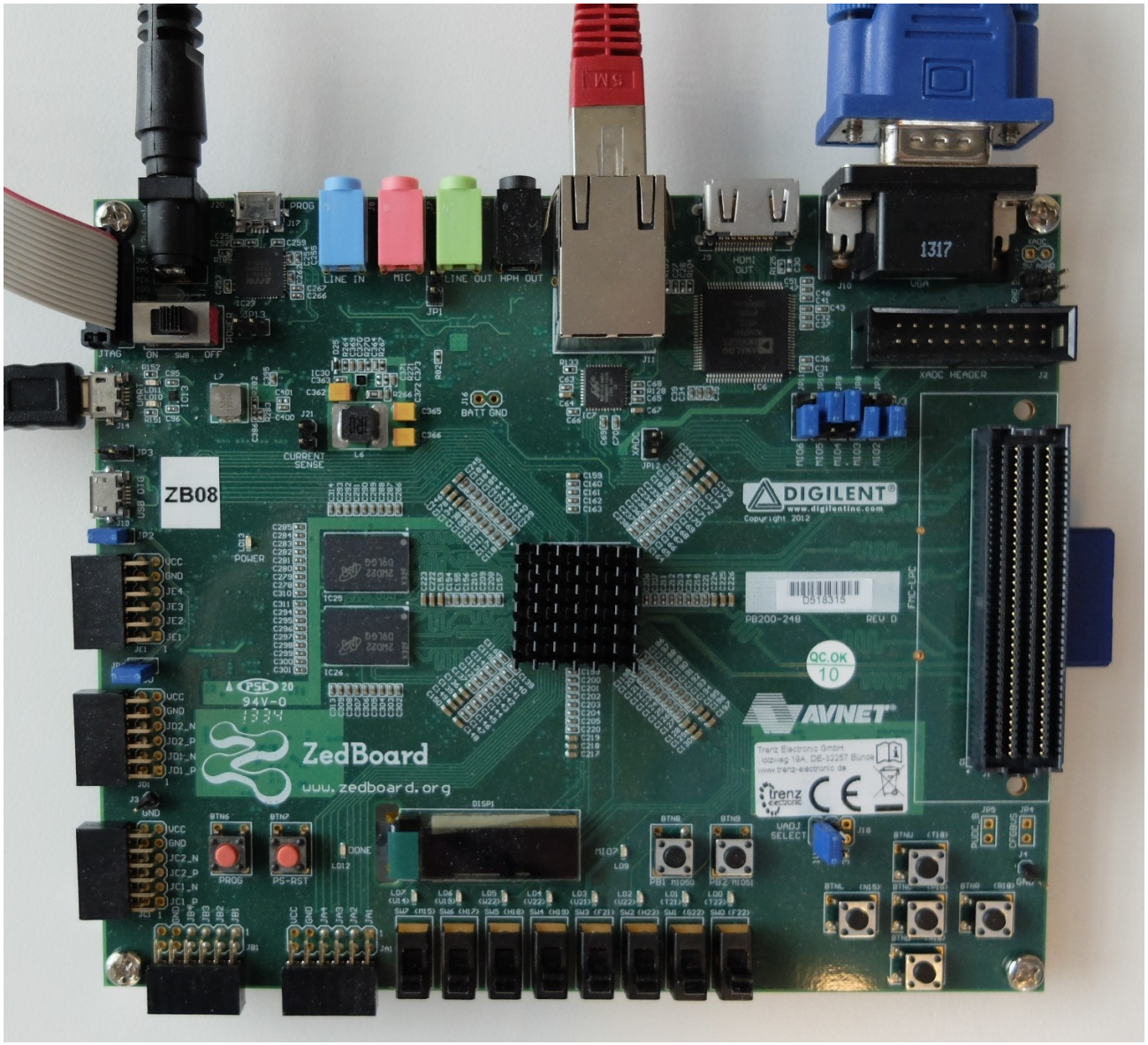
Memory Tracing

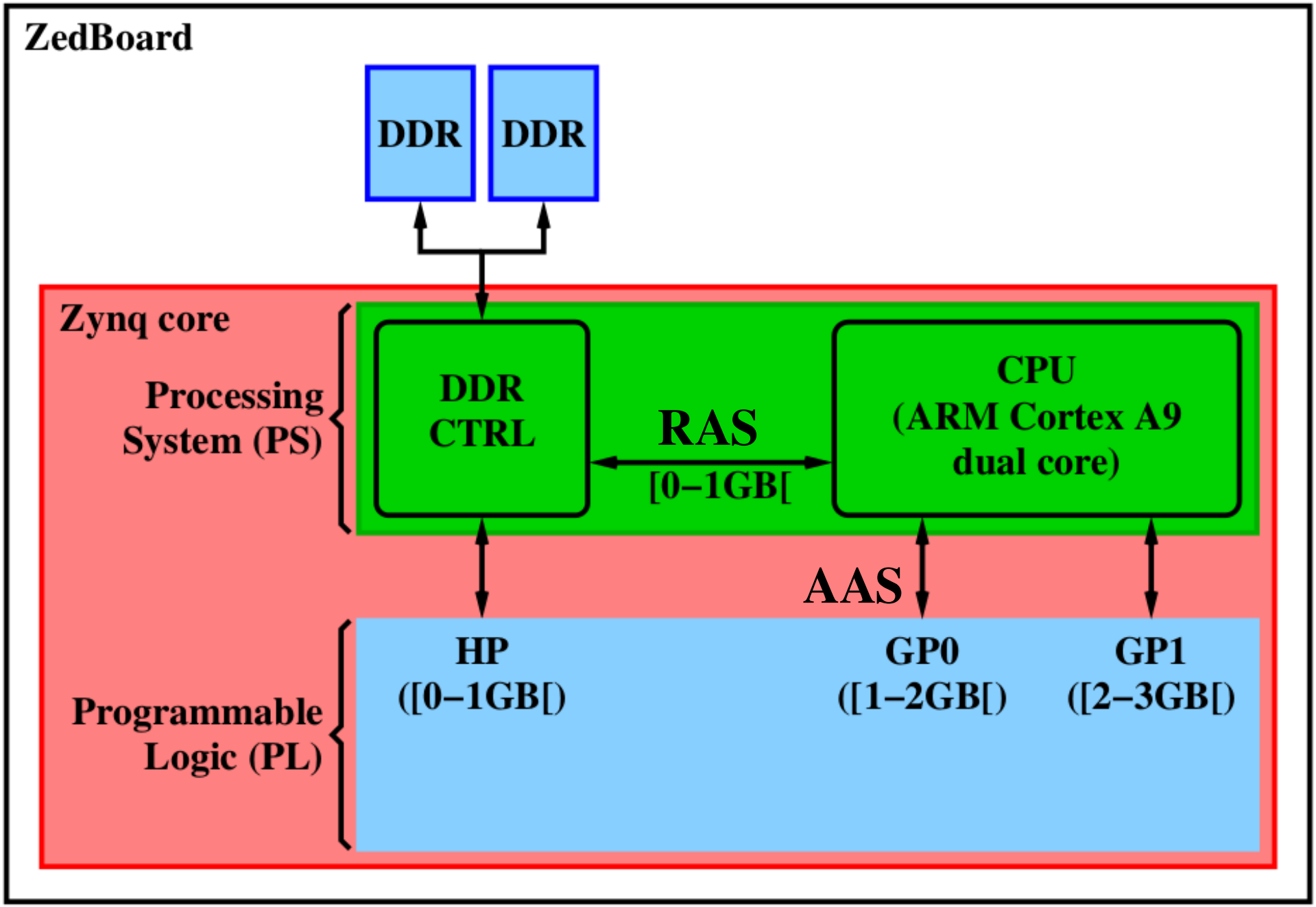
- Analysis and categorization of malware
- Debugging
- Hardware support for improved performance

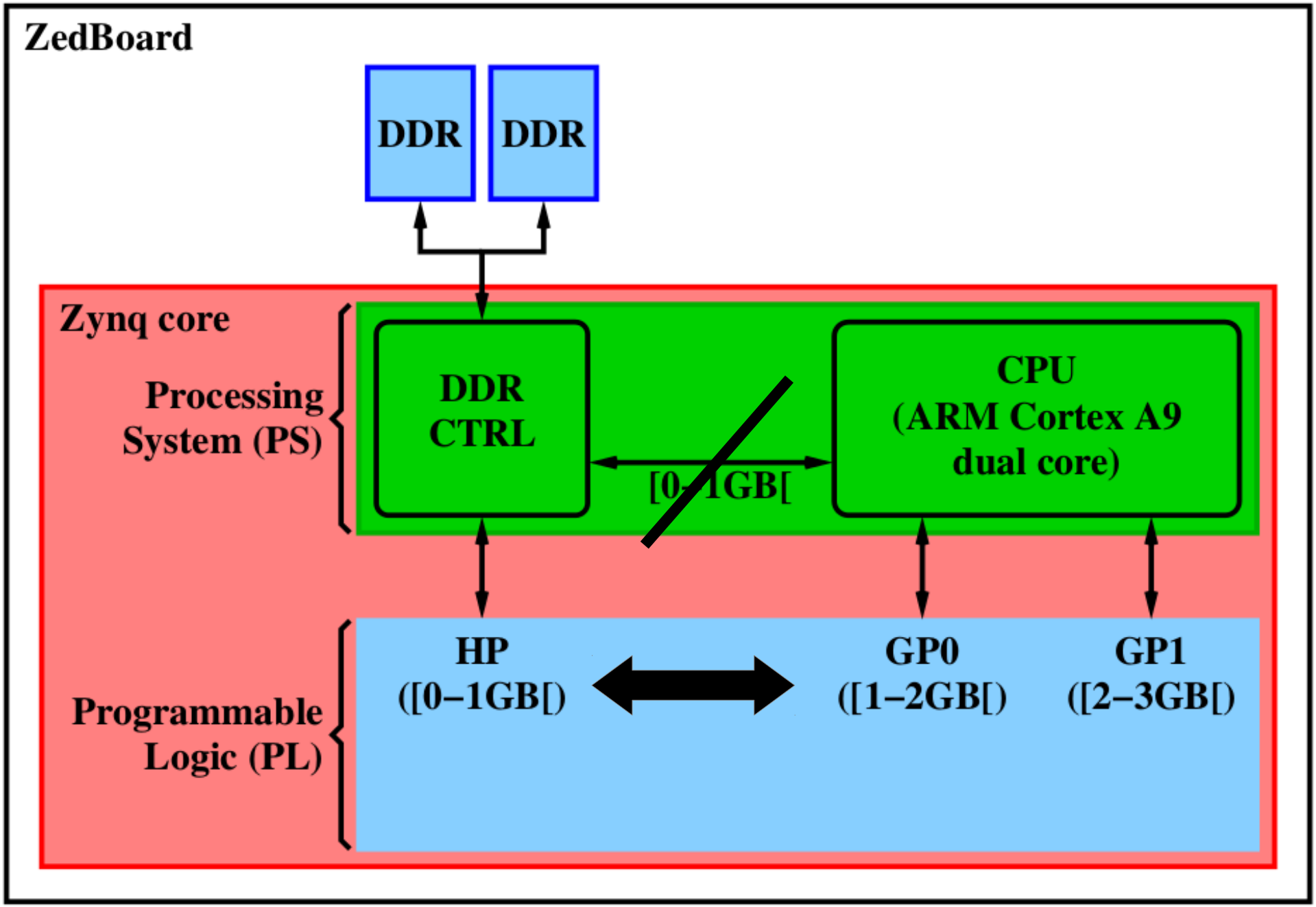
SecBus Project

- How to secure System-on-chip?
- Attacker can access memory bus to external memory
- Platform for demonstration and validation

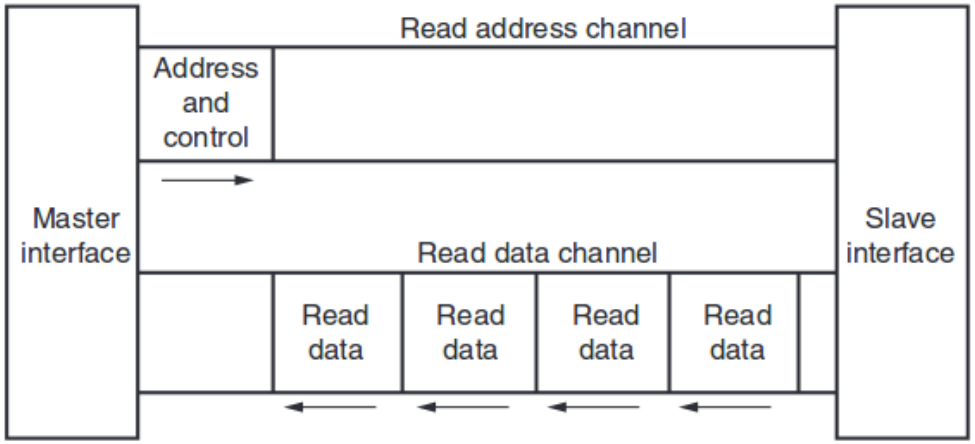
Xilinx Zynq-based Zedboard by Avnet

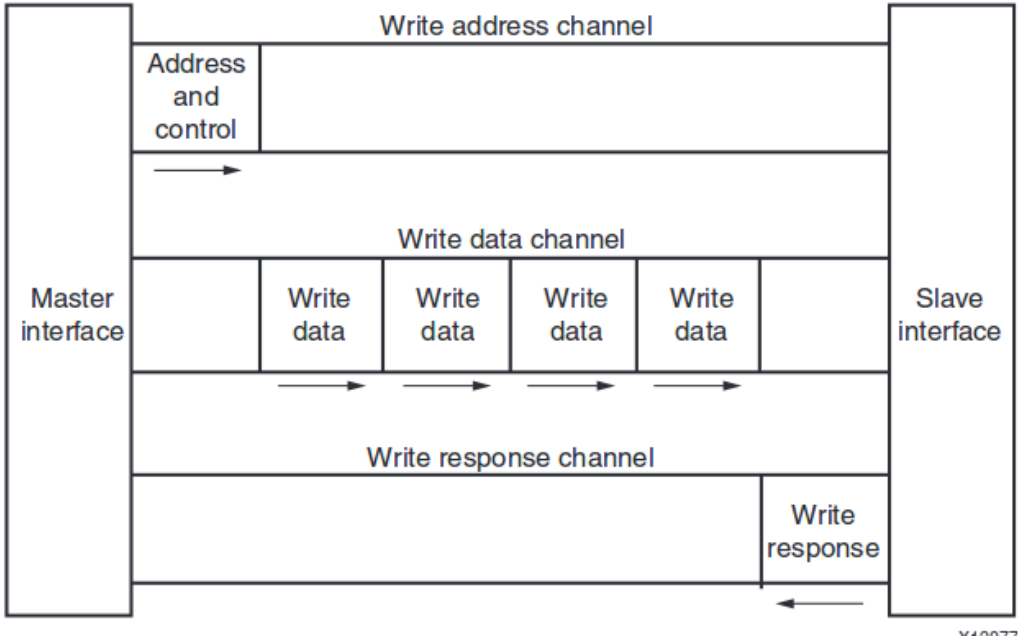



ZedBoard
DDR
DDR
Zynq core
Processing System (PS)
DDR CTRL
RAS
[0–1GB[
CPU (ARM Cortex A9 dual core)
AAS
Programmable Logic (PL)
HP ([0–1GB[)
GP0 ([1–2GB[)
GP1 ([2–3GB[)

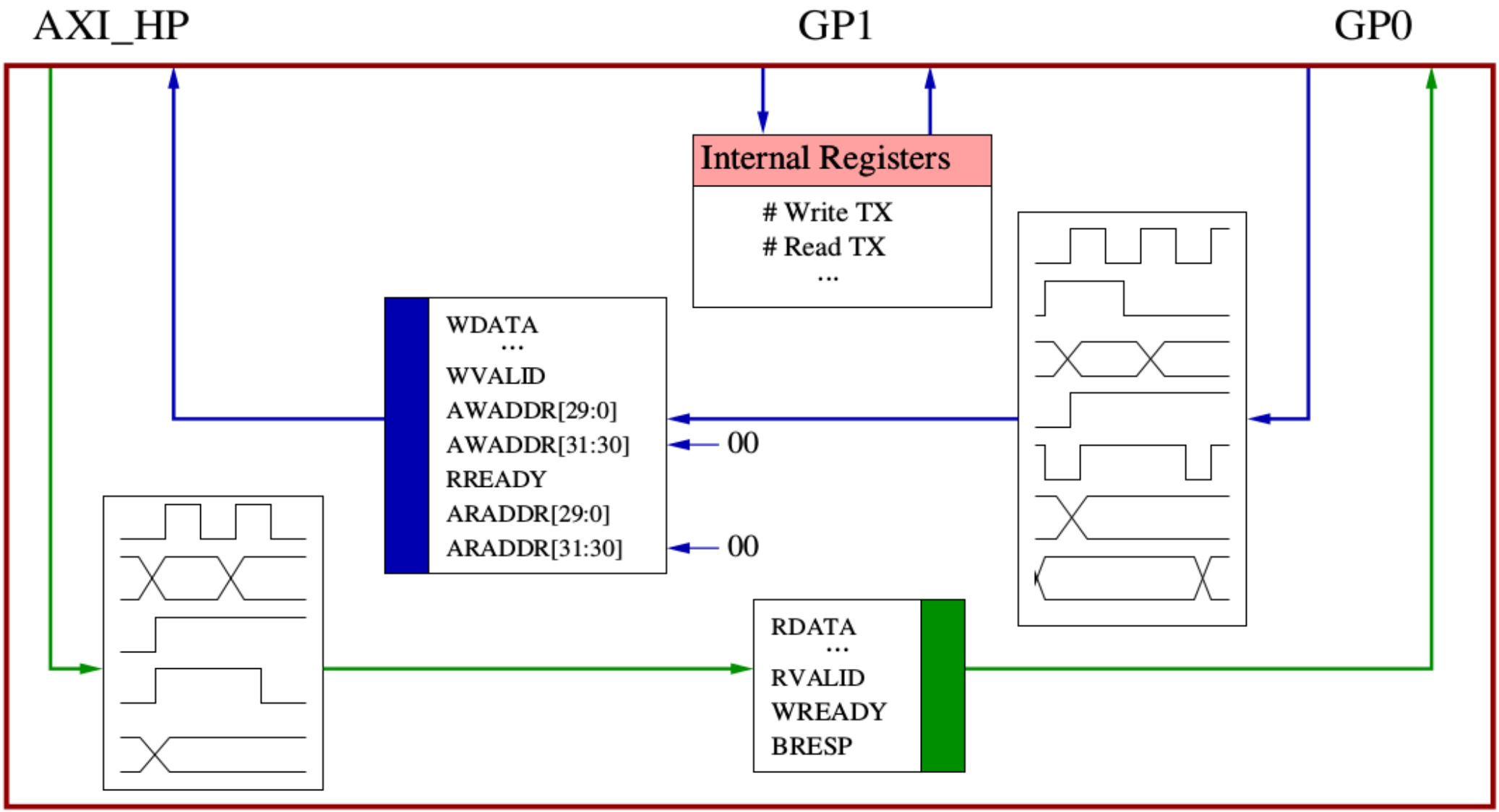

ZedBoard
DDR
DDR
Zynq core
Processing System (PS)
DDR CTRL
CPU (ARM Cortex A9 dual core)
[0–1GB[
Programmable Logic (PL)
HP ([0–1GB[)
GP0 ([1–2GB[)
GP1 ([2–3GB[)

AXI Protocol





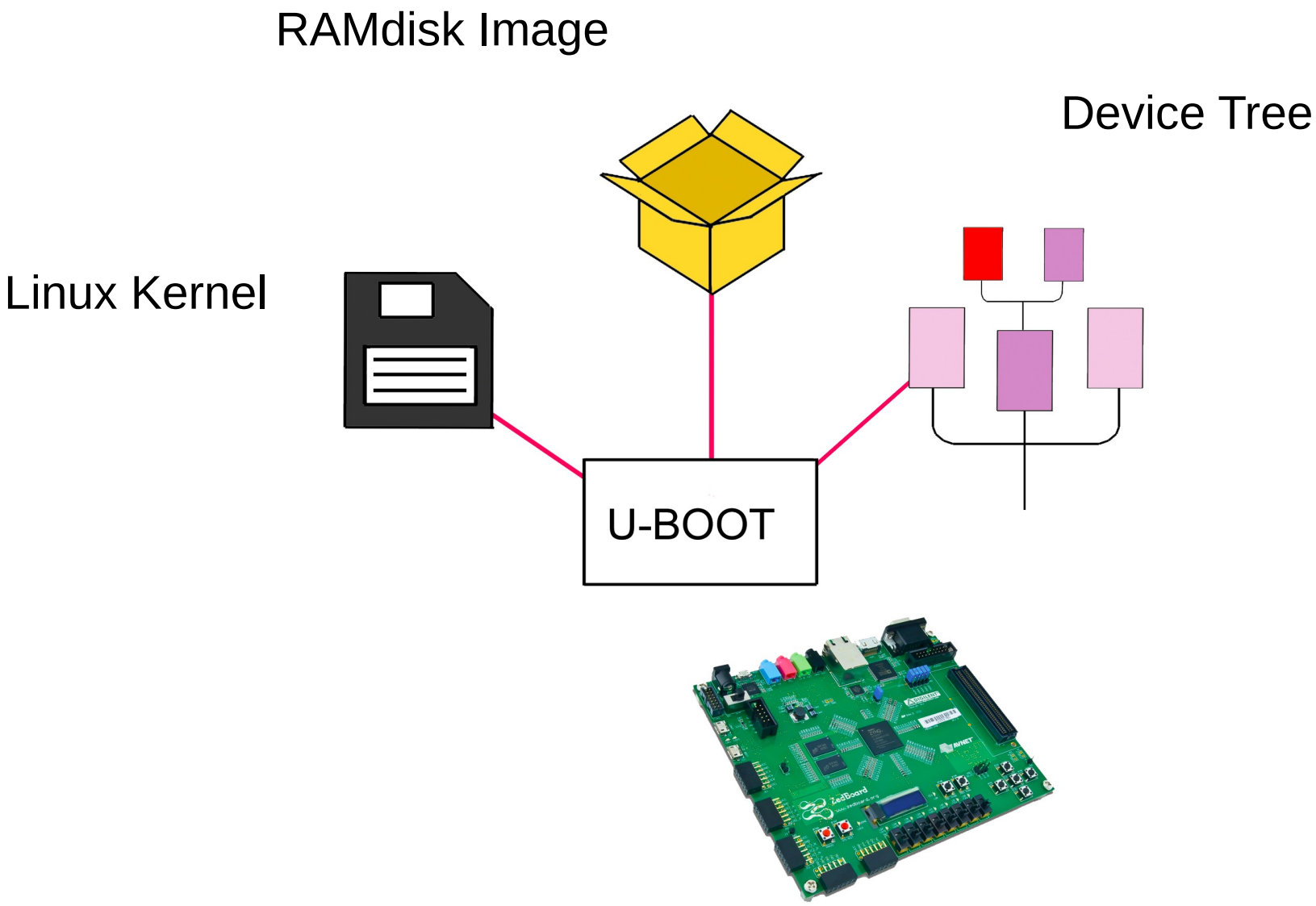
http://www.xilinx.com/support/documentation/ip_documentation/axi_ref_guide/v13_4/ug761_axi_reference_guide.pdf


AXI_HP
GP1
GP0
Internal Registers
Write TX
Read TX
...
WDATA
...
WVALID
AWADDR[29:0]
AWADDR[31:30]
RREADY
ARADDR[29:0]
ARADDR[31:30]
00
00
RDATA
...
RVALID
WREADY
BRESP

Implementation

Linux in Alternate Address Space

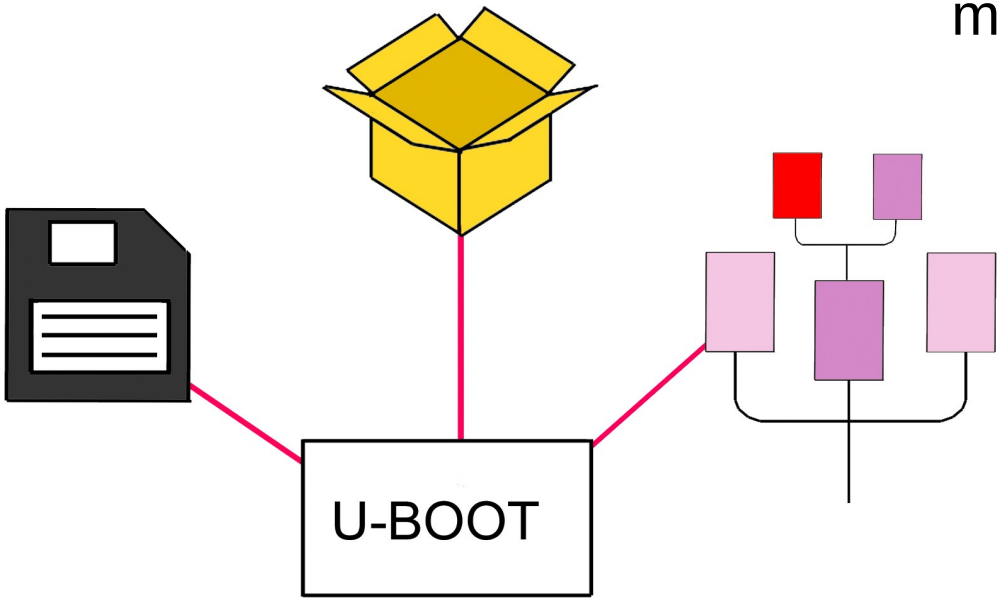
Software Configuration



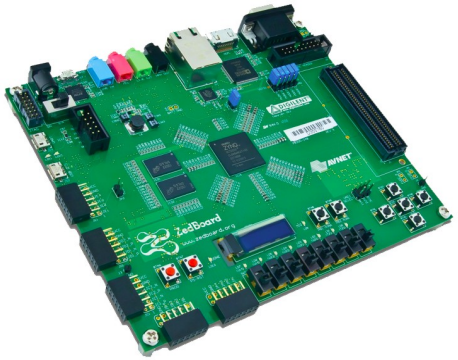
→ Alternate Address Space

Entry Point

Linux, usable-memory



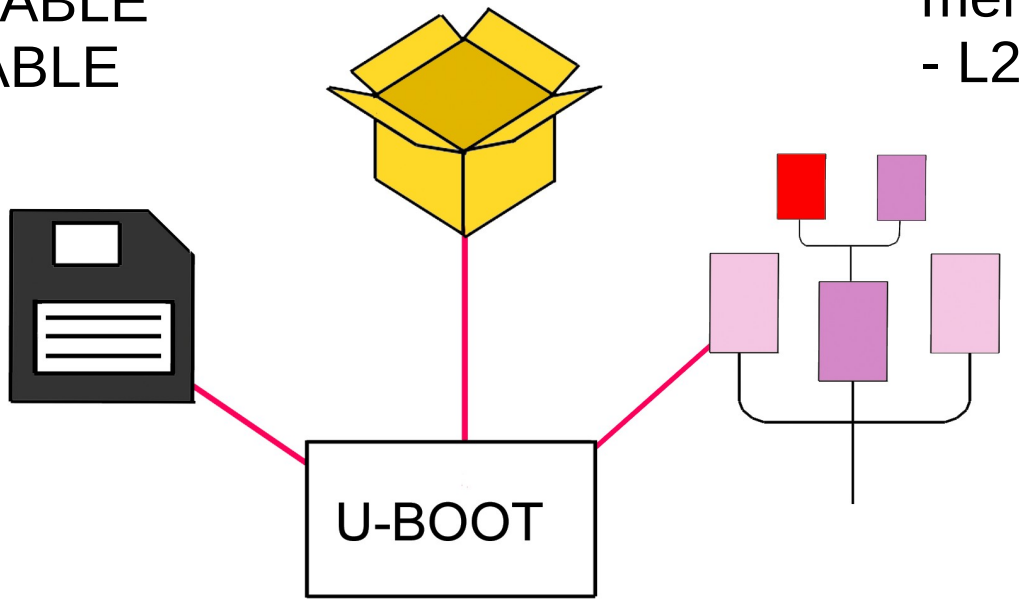
FDT_HIGH
Load Addresses



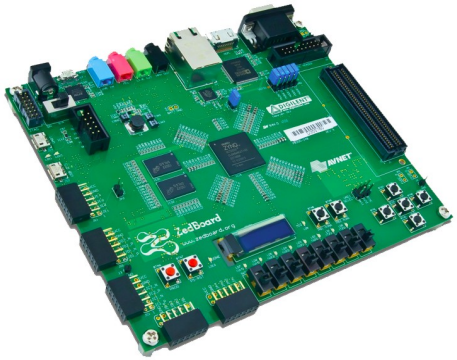
- Caching

Entry Point
INSTR_CACHE_DISABLE
DATA_CACHE_DISABLE
Flush cache

Linux, usable-memory
- L2 cache

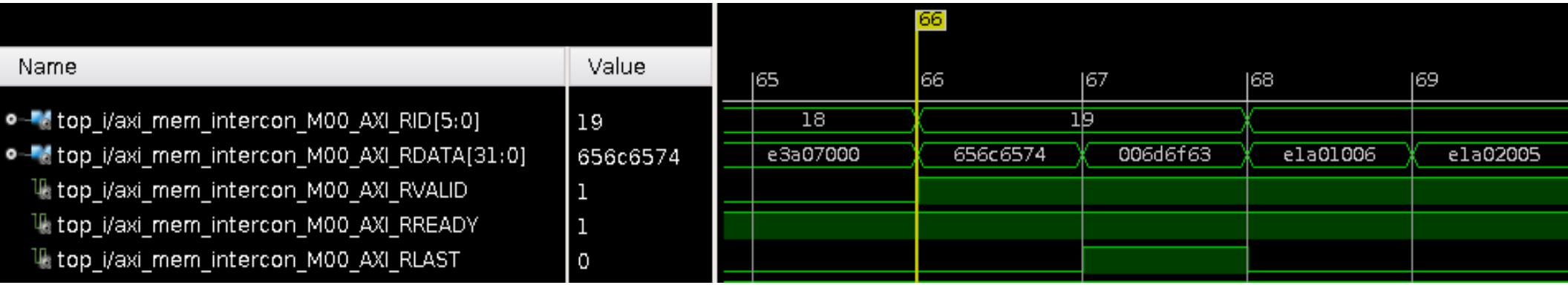


FDT_HIGH
Load Addresses

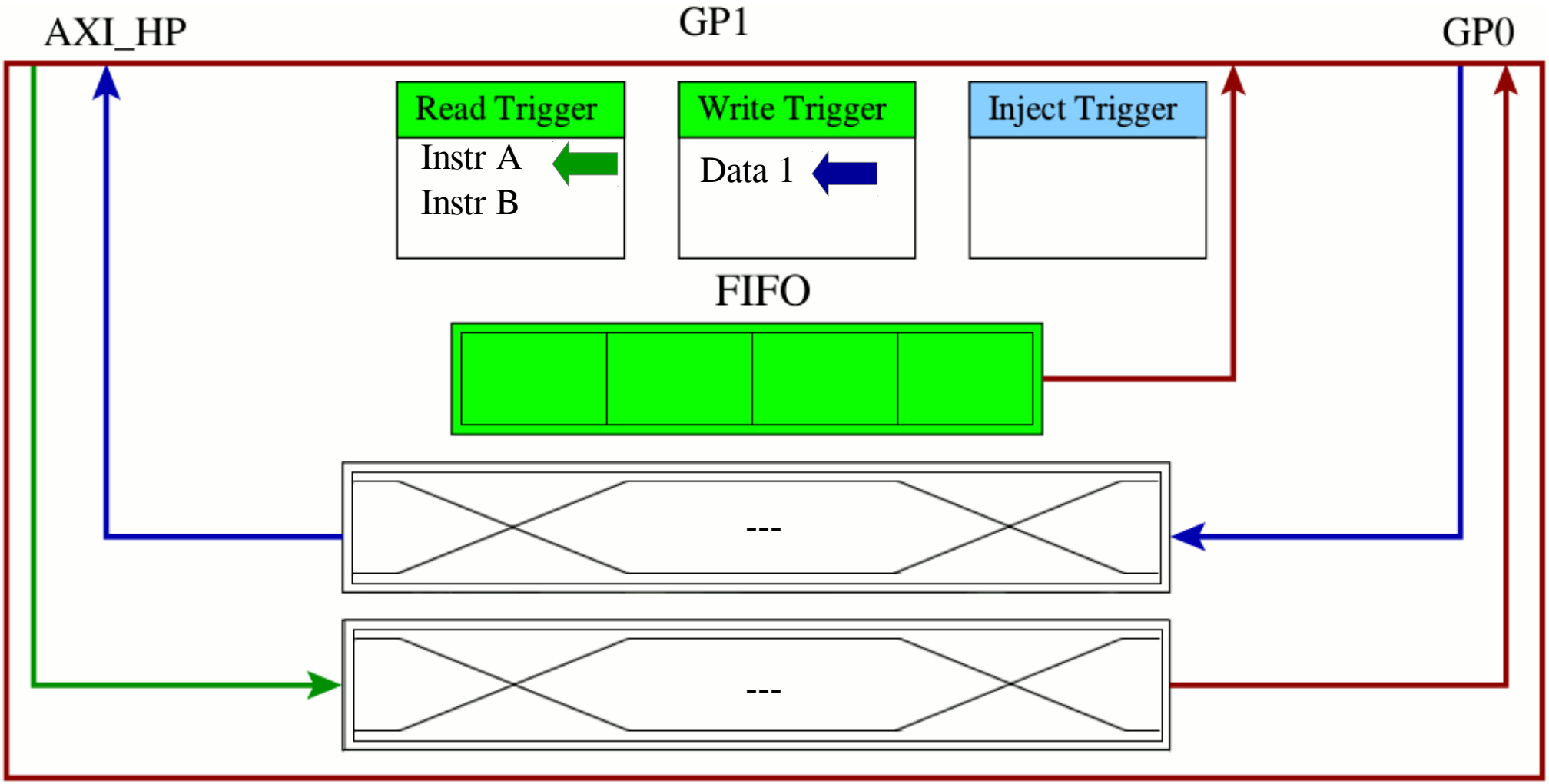


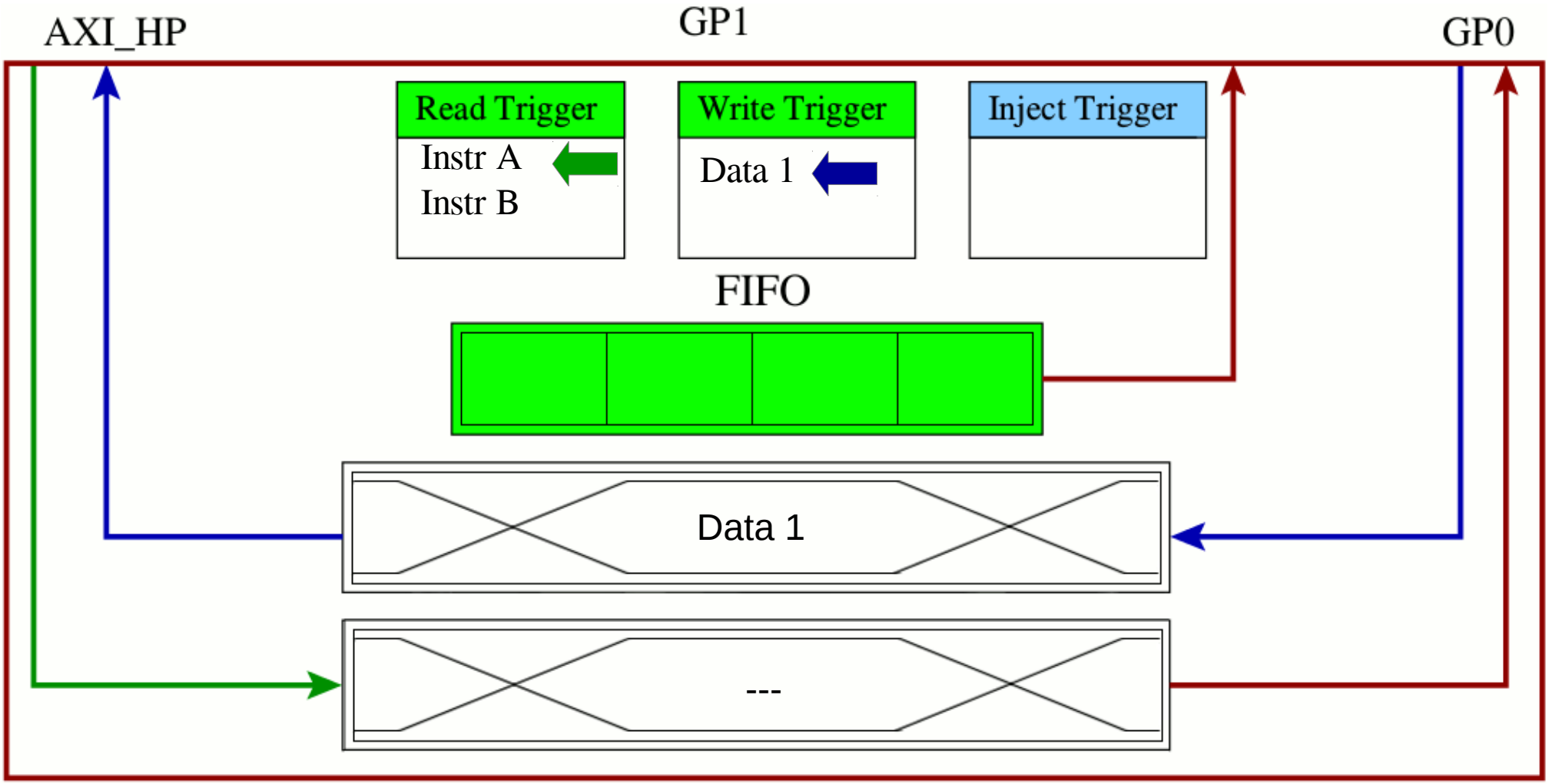
Experiments

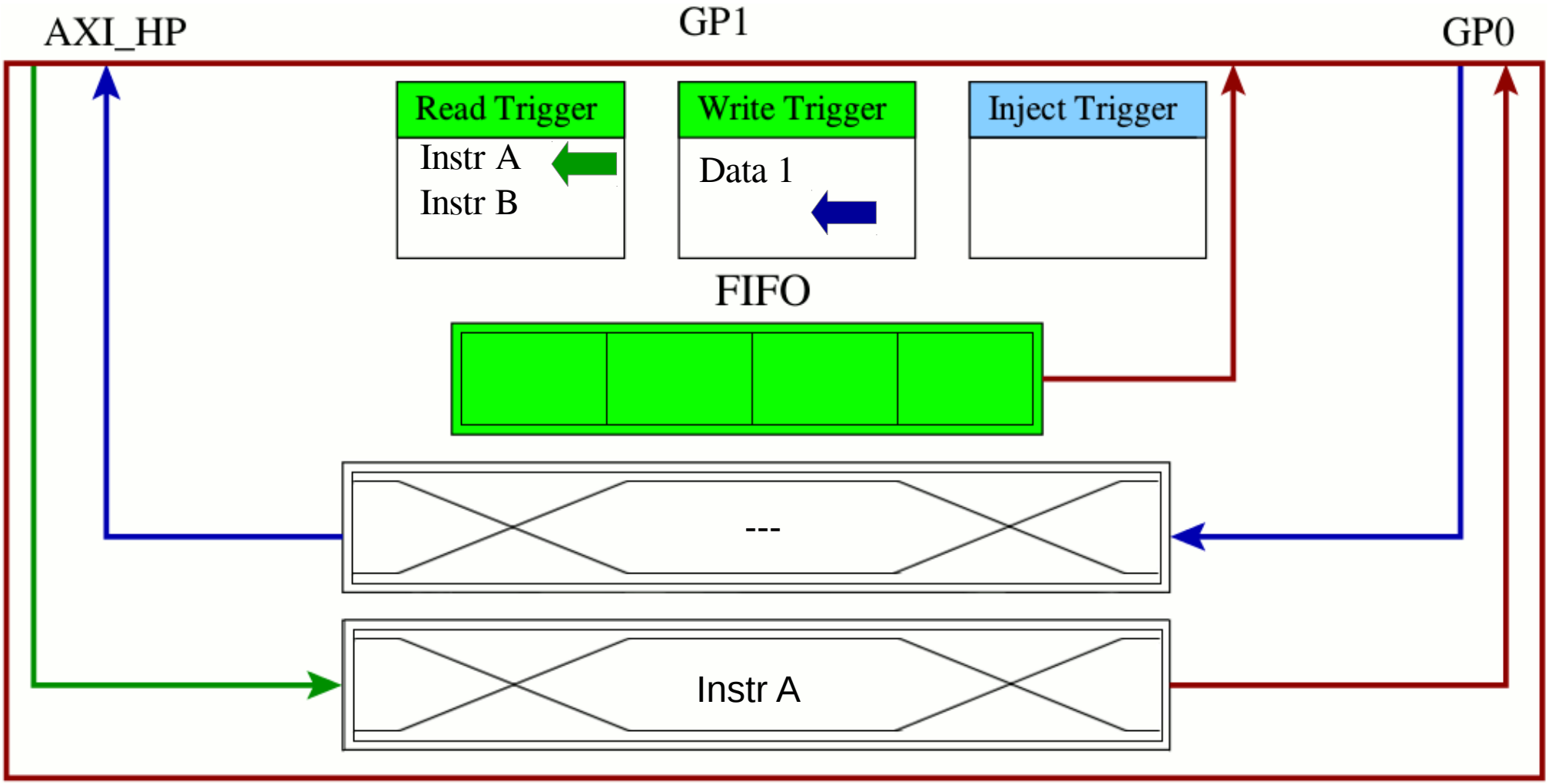
SSH



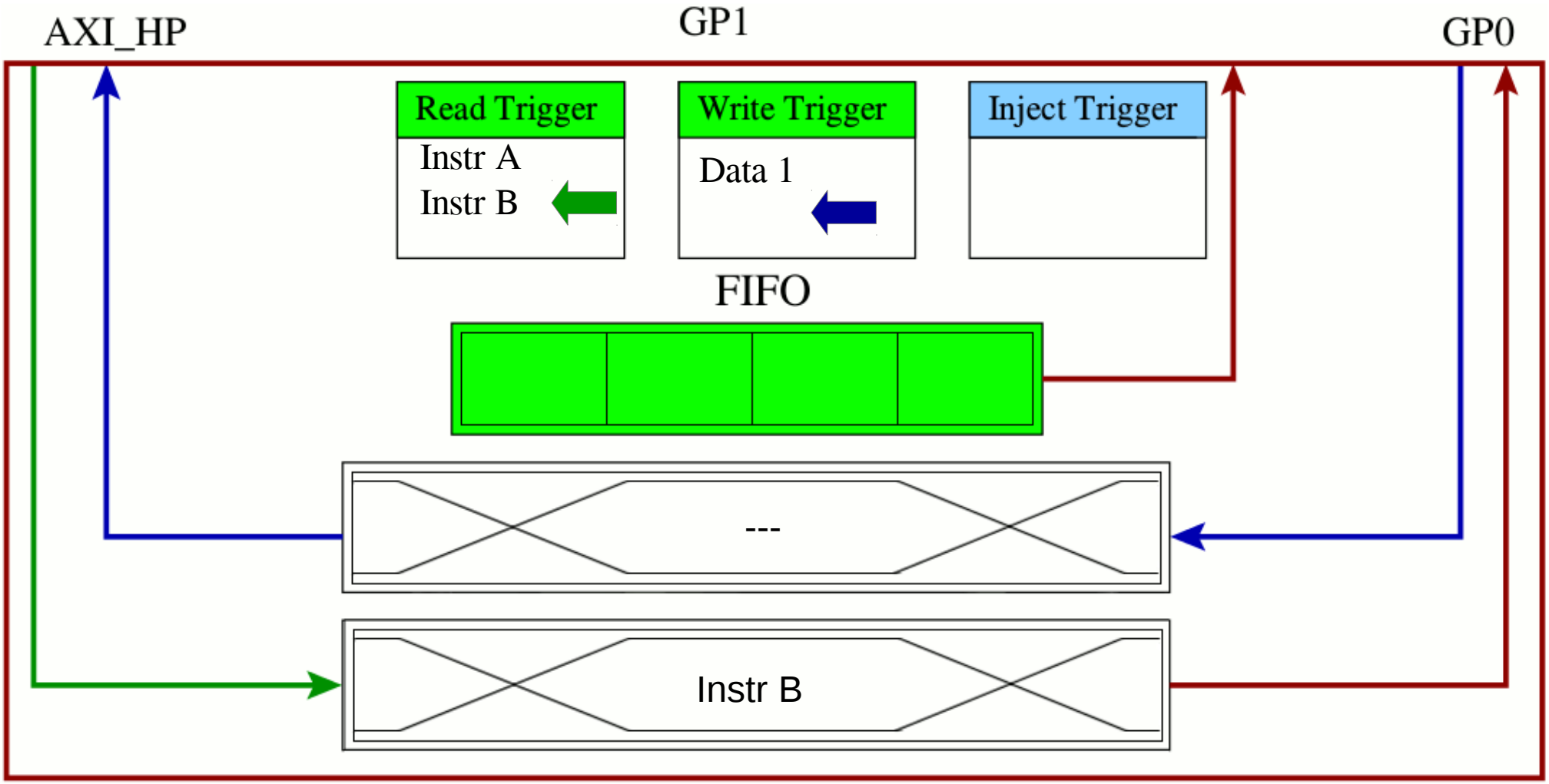
74: t
65: e
6C: l
65: e
63: c
6F: o
6D: m


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO

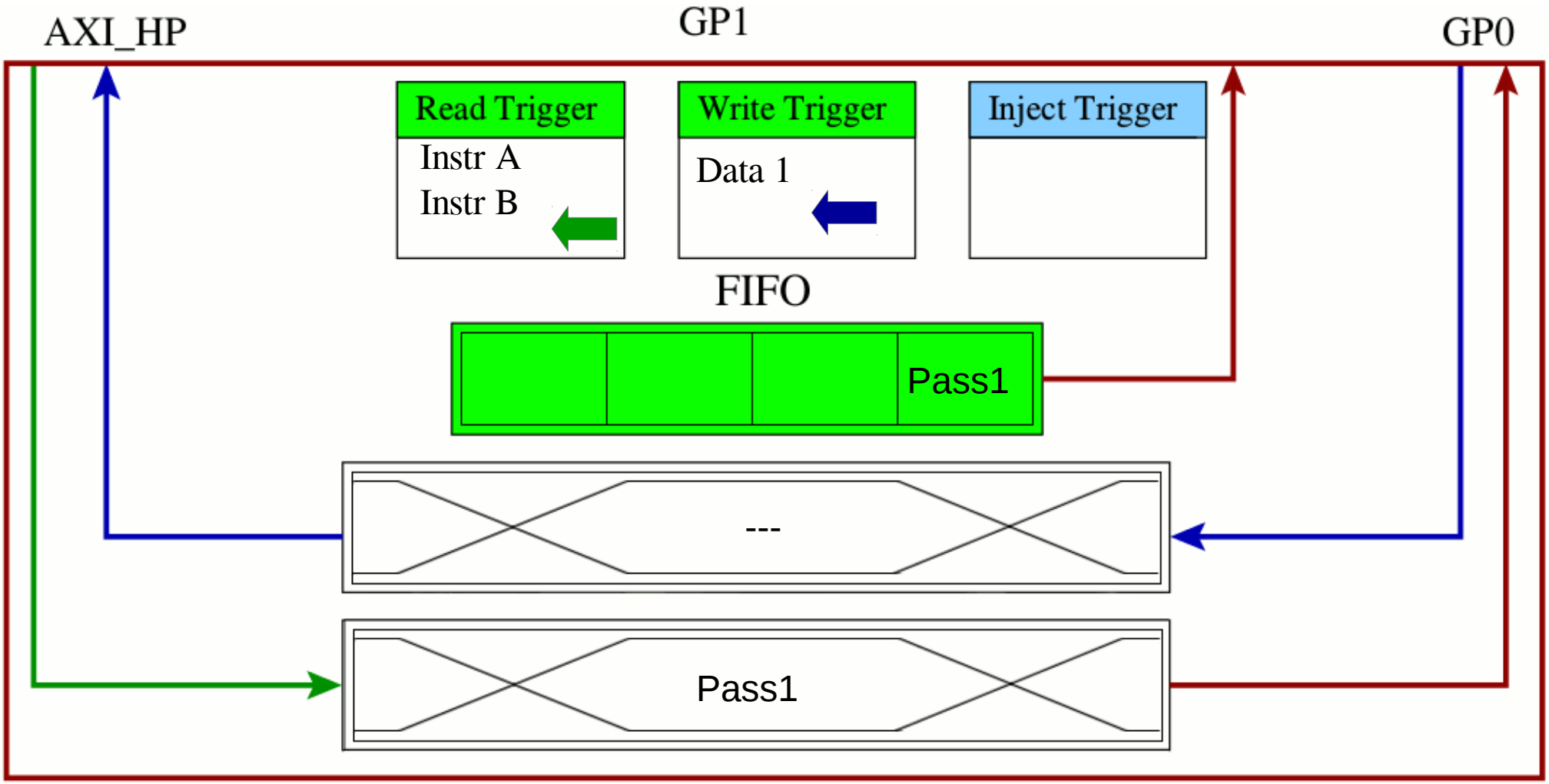

AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO
Data 1


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO

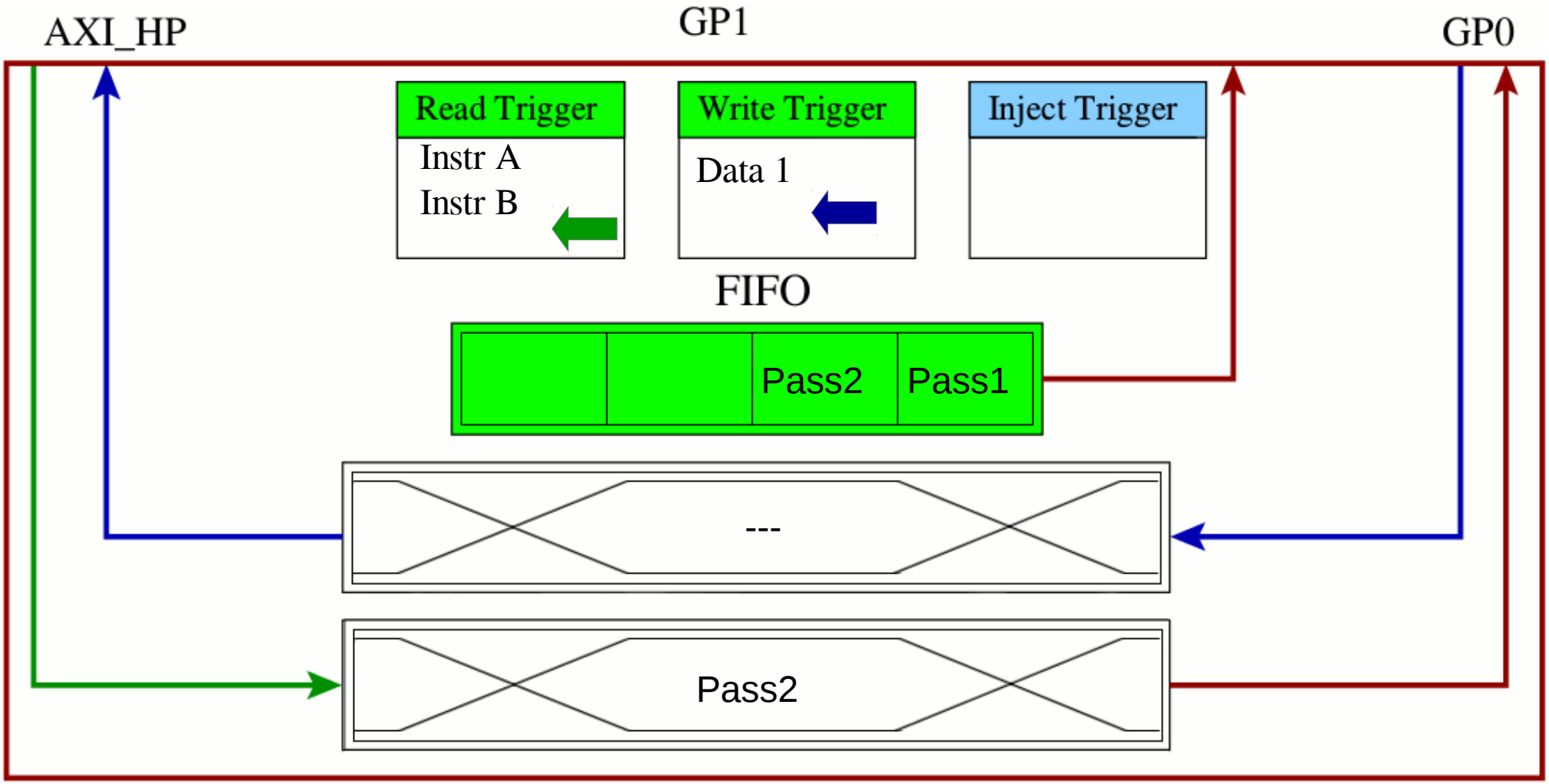
Instr A


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO

Instr B


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO
Pass1

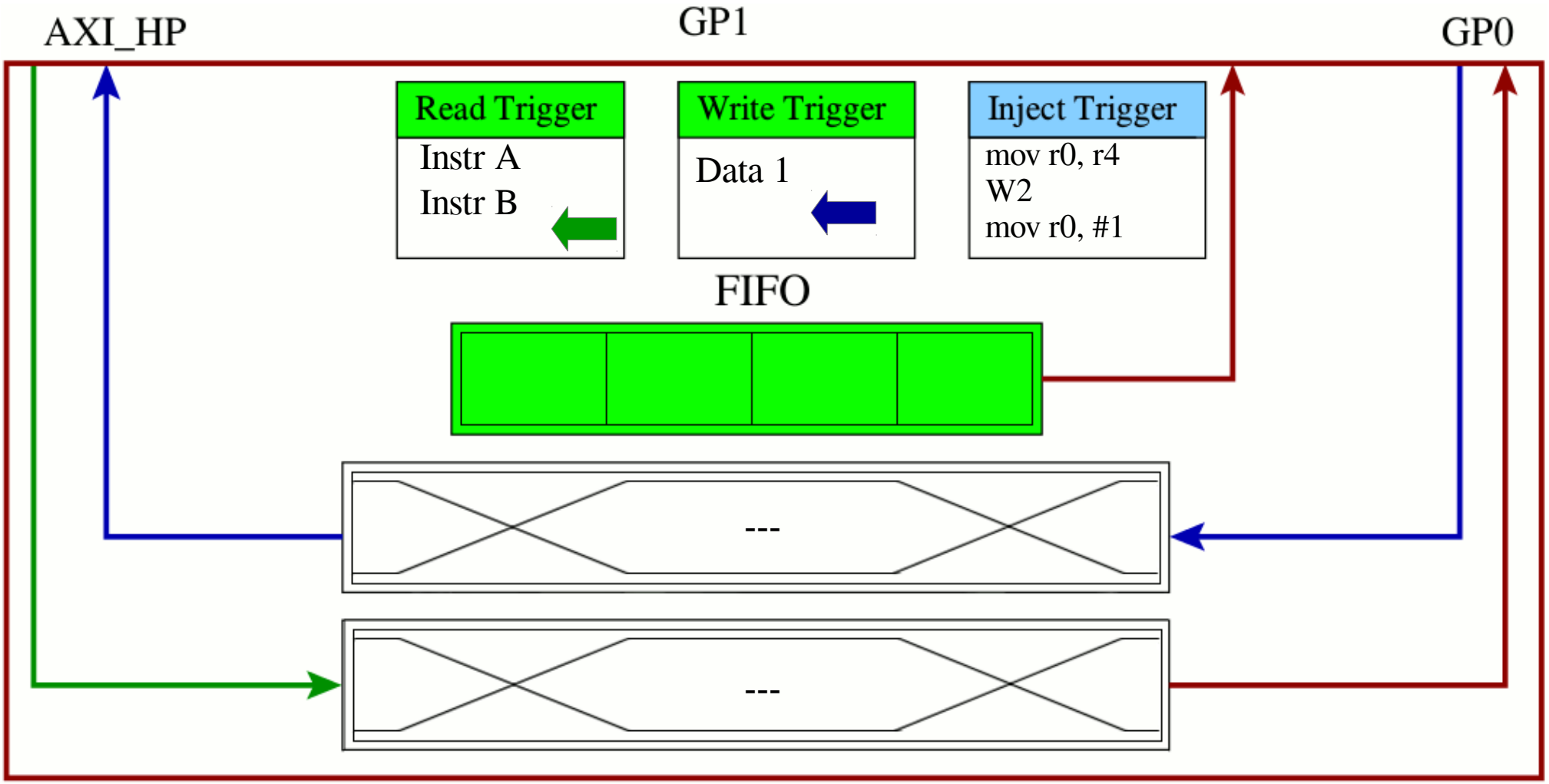
Pass1

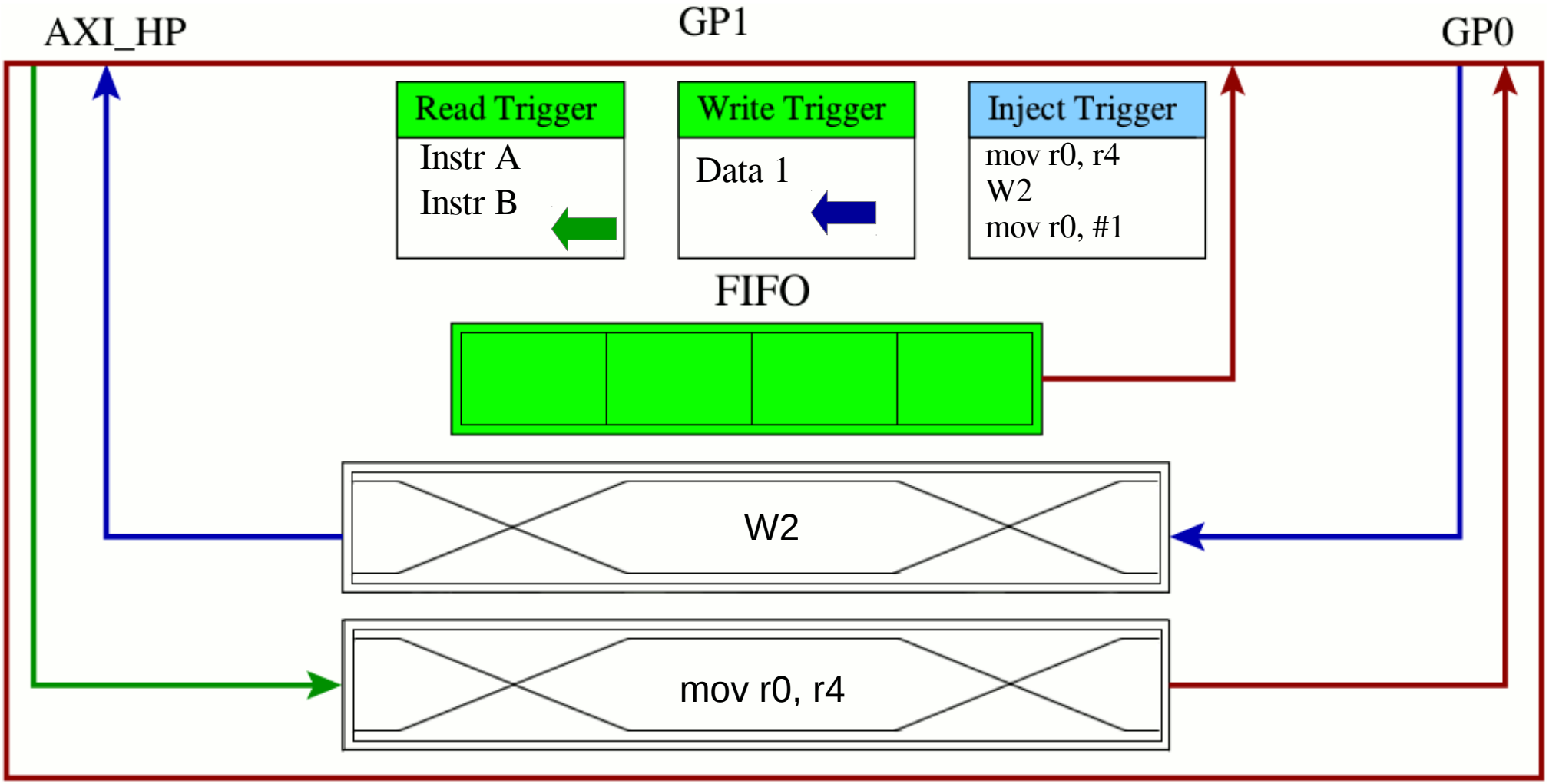

AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
FIFO
Pass2
Pass1

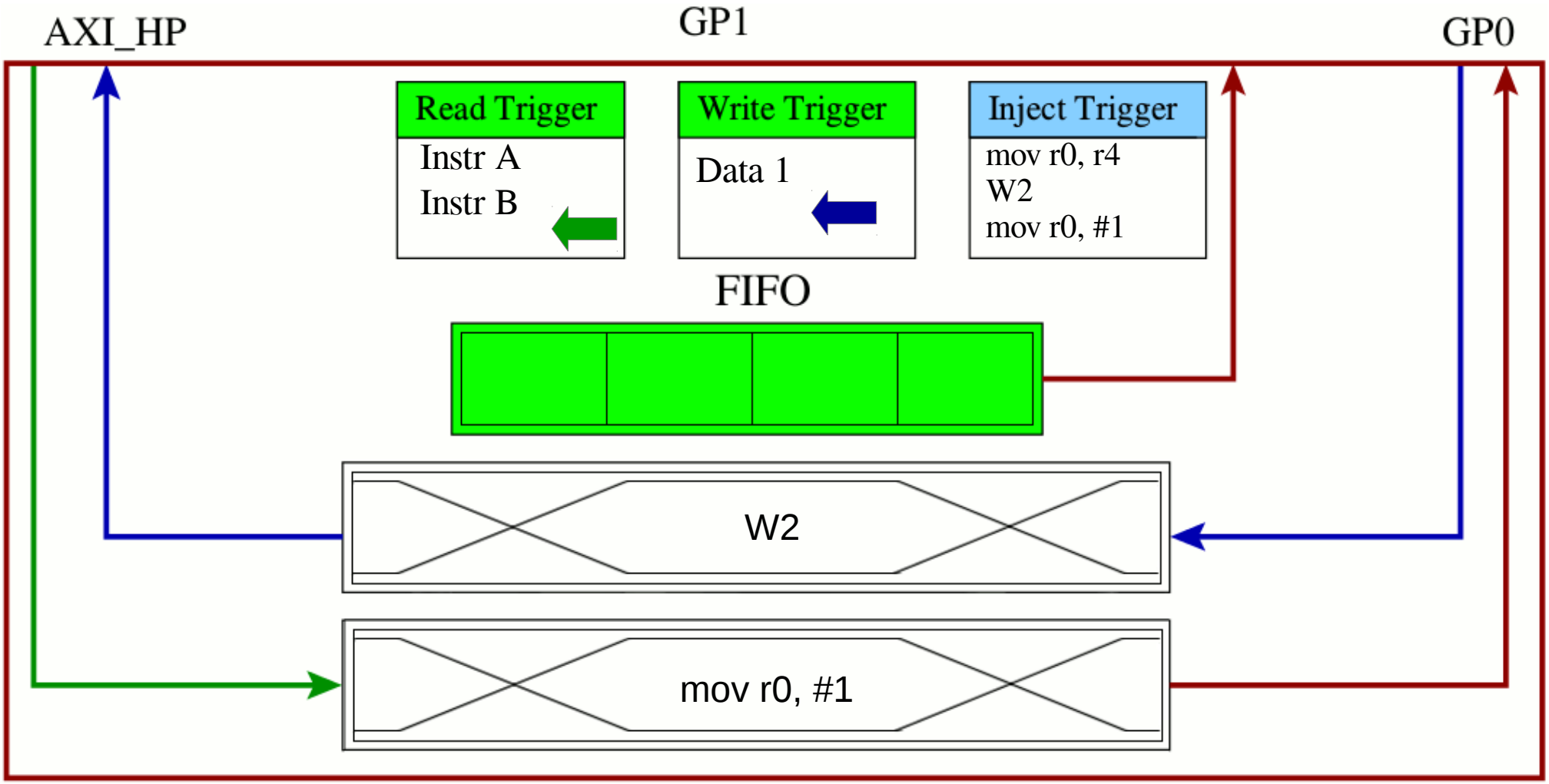
Pass2

Login

- BusyBox
- Implementation
  - String Compare
  - Result in r4
  - mov r0, r4 ← mov r0, #1


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
mov r0, r4
W2
mov r0, #1
FIFO

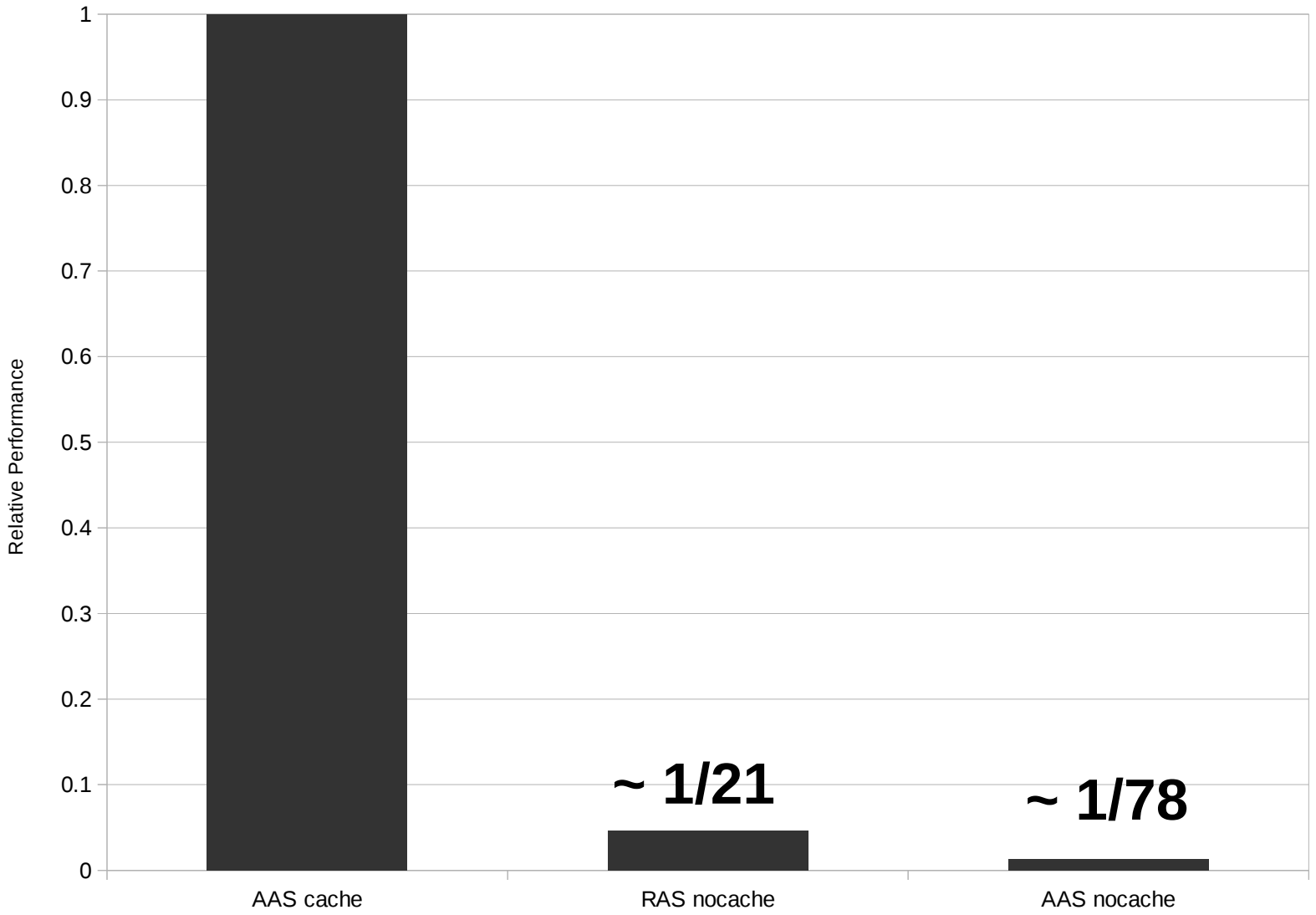

AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
mov r0, r4
W2
mov r0, #1
FIFO
W2
mov r0, r4


AXI_HP
GP1
GP0
Read Trigger
Instr A
Instr B
Write Trigger
Data 1
Inject Trigger
mov r0, r4
W2
mov r0, #1
FIFO
W2
mov r0, #1

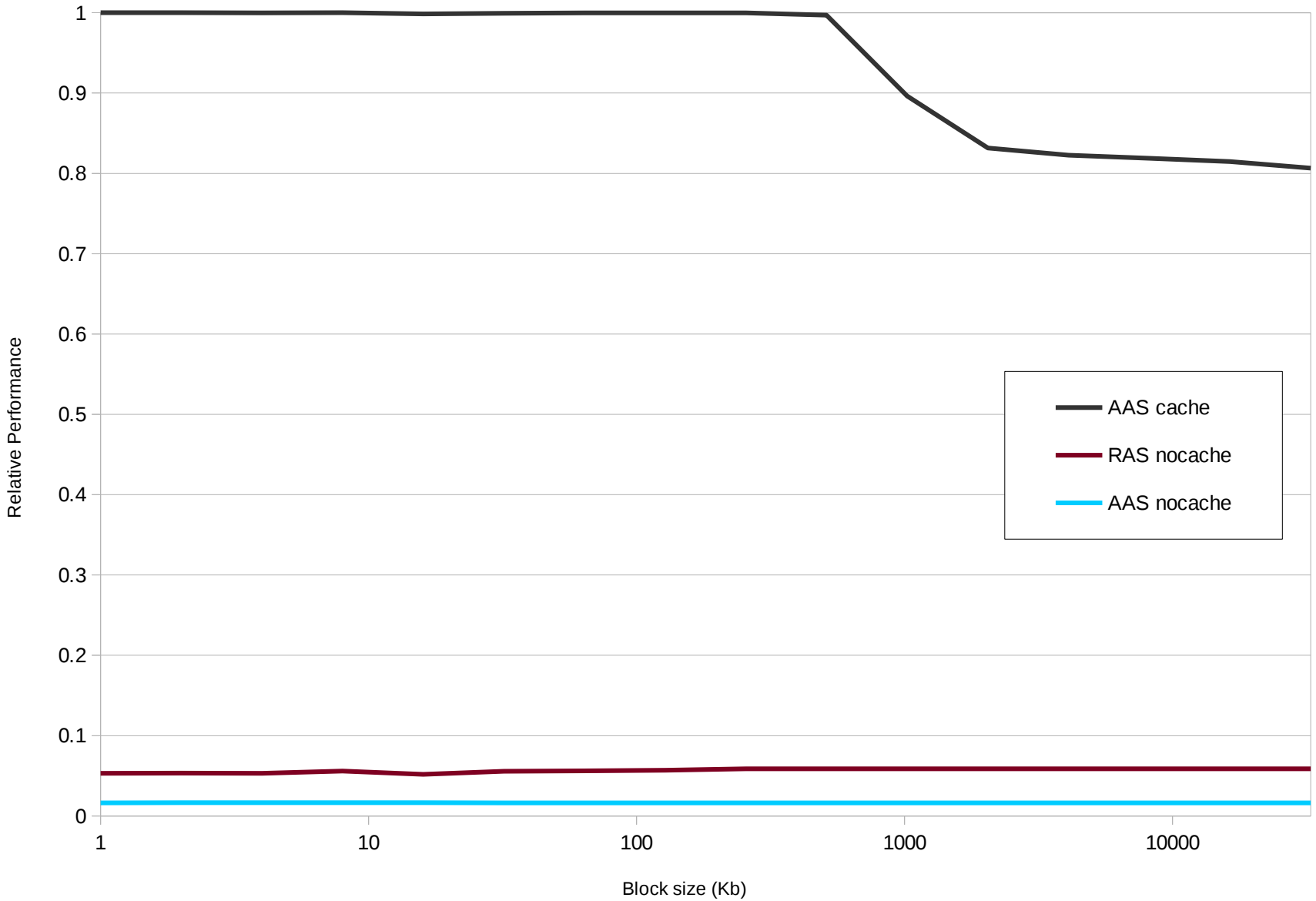
Performance

- Dhrystone
  - Integer benchmark
- RAMspeed
  - Read/Write blocks

Dhrystone


Relative Performance
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
~ 1/21
~ 1/78
AAS cache
RAS nocache
AAS nocache

RAMSpeed



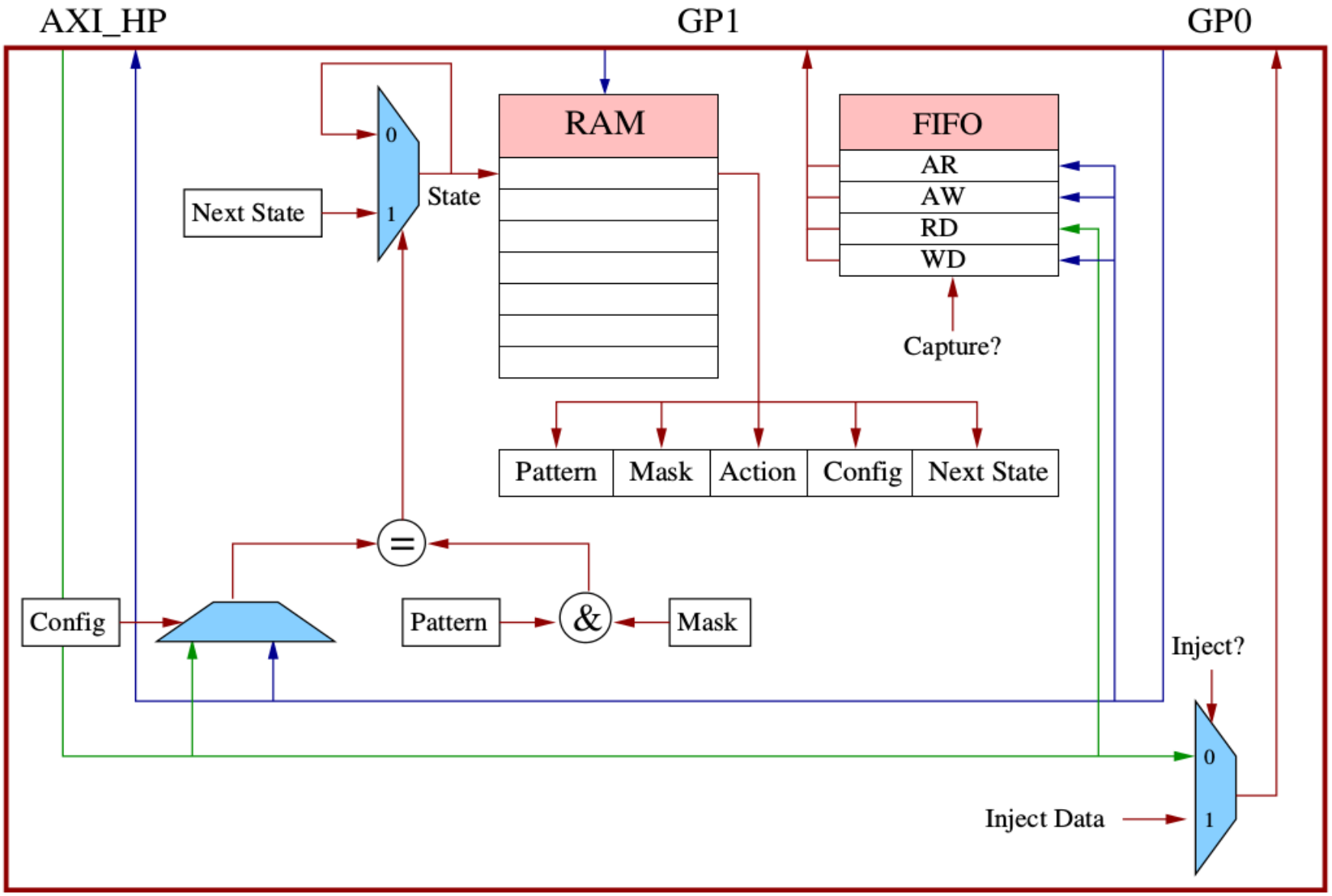
~ 1/1.23

~ 1/17

~ 1/62

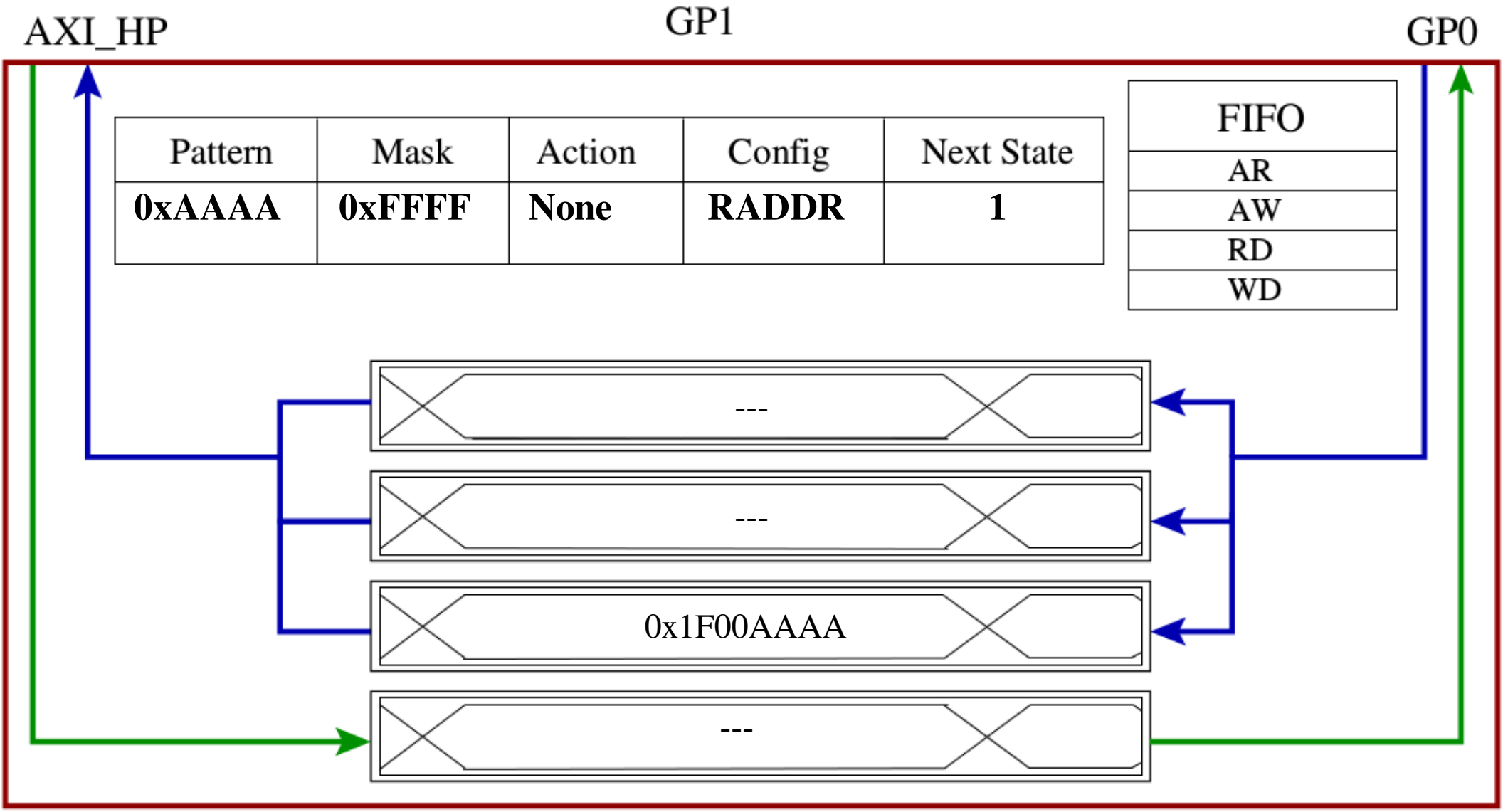
Memory Access Monitor 2.0

- Cache-Enabled
- More Channels
- Precise Capture
- Multiple Injection Patterns

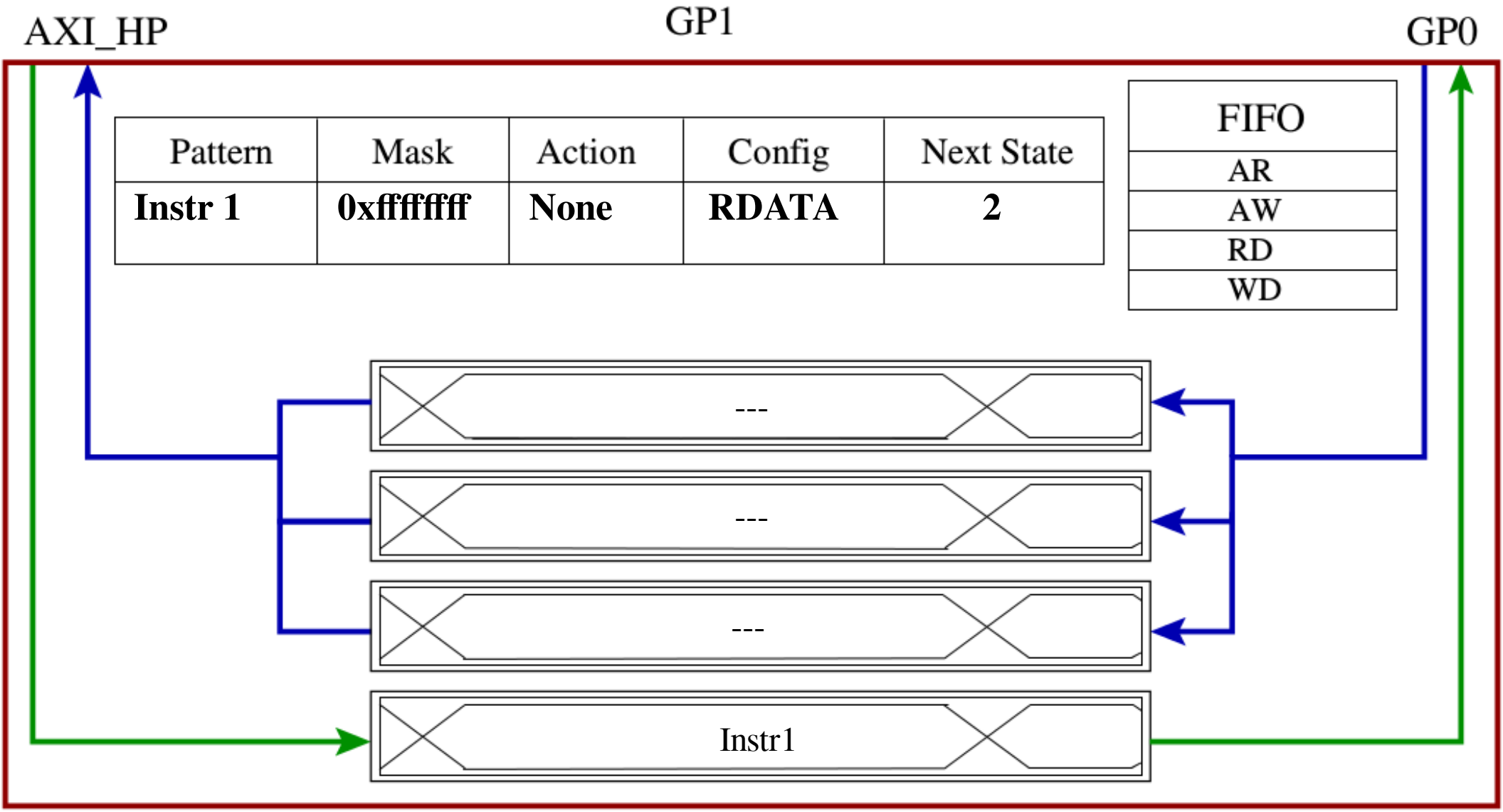

AXI_HP
GP1
GP0
RAM
FIFO
AR
AW
RD
WD
Capture?
0
1
Next State
State
Pattern
Mask
Action
Config
Next State
=
Config
Pattern
&
Mask
Inject?
0
1
Inject Data

SSH v2.0

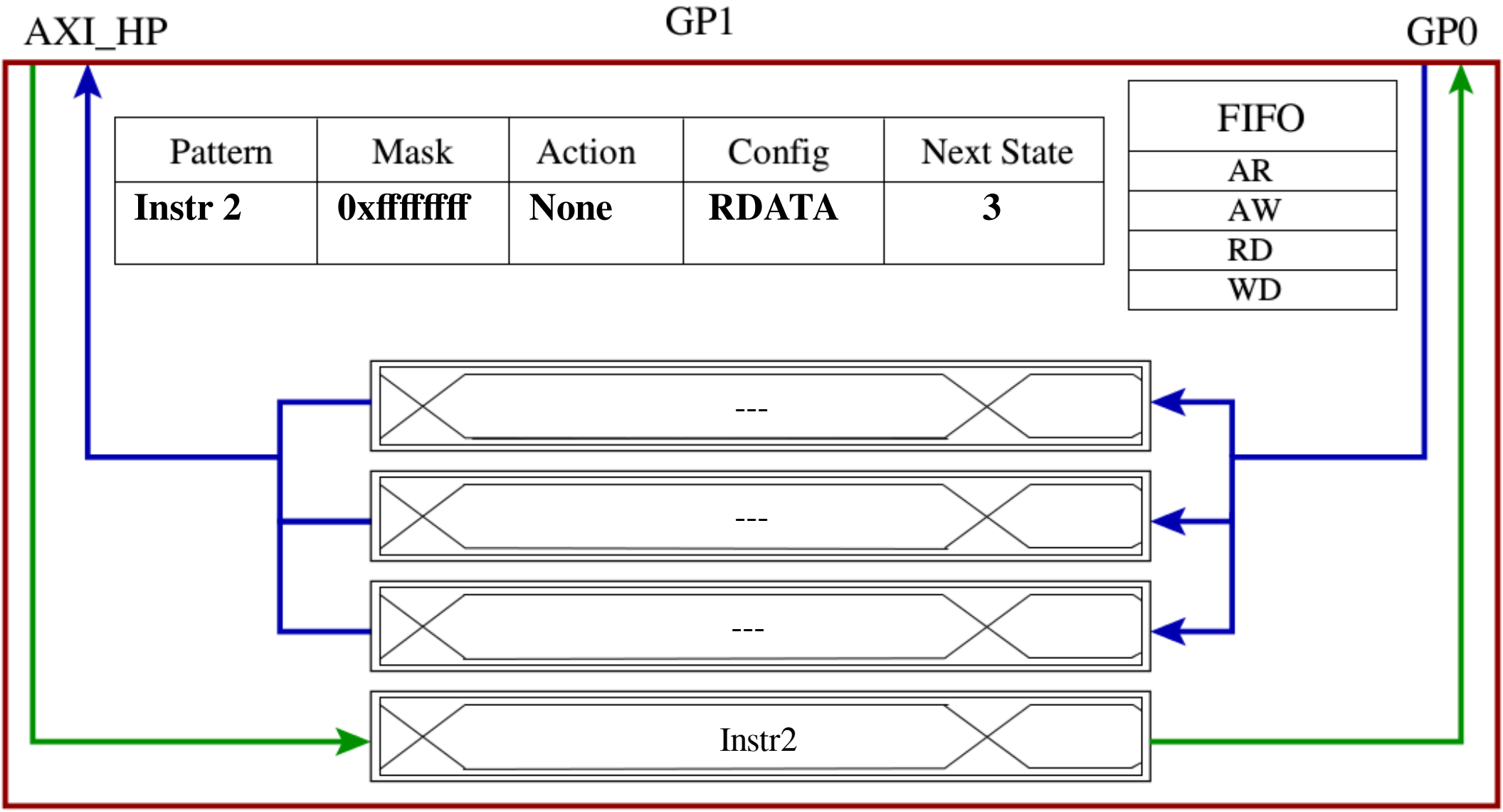
State	Pattern	Mask	Action	Config	Next State
0	Address	0xFFFF	None	RADDR	1
1	Instr 1	0xFFFFFFFF	None	RDATA	2
2	Instr 2	0xFFFFFFFF	None	RDATA	3
3	0x0	0xFFFFFFFF	Capture	RDATA	0


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
0xAAAA
0xFFFF
None
RADDR
1
FIFO
AR
AW
RD
WD

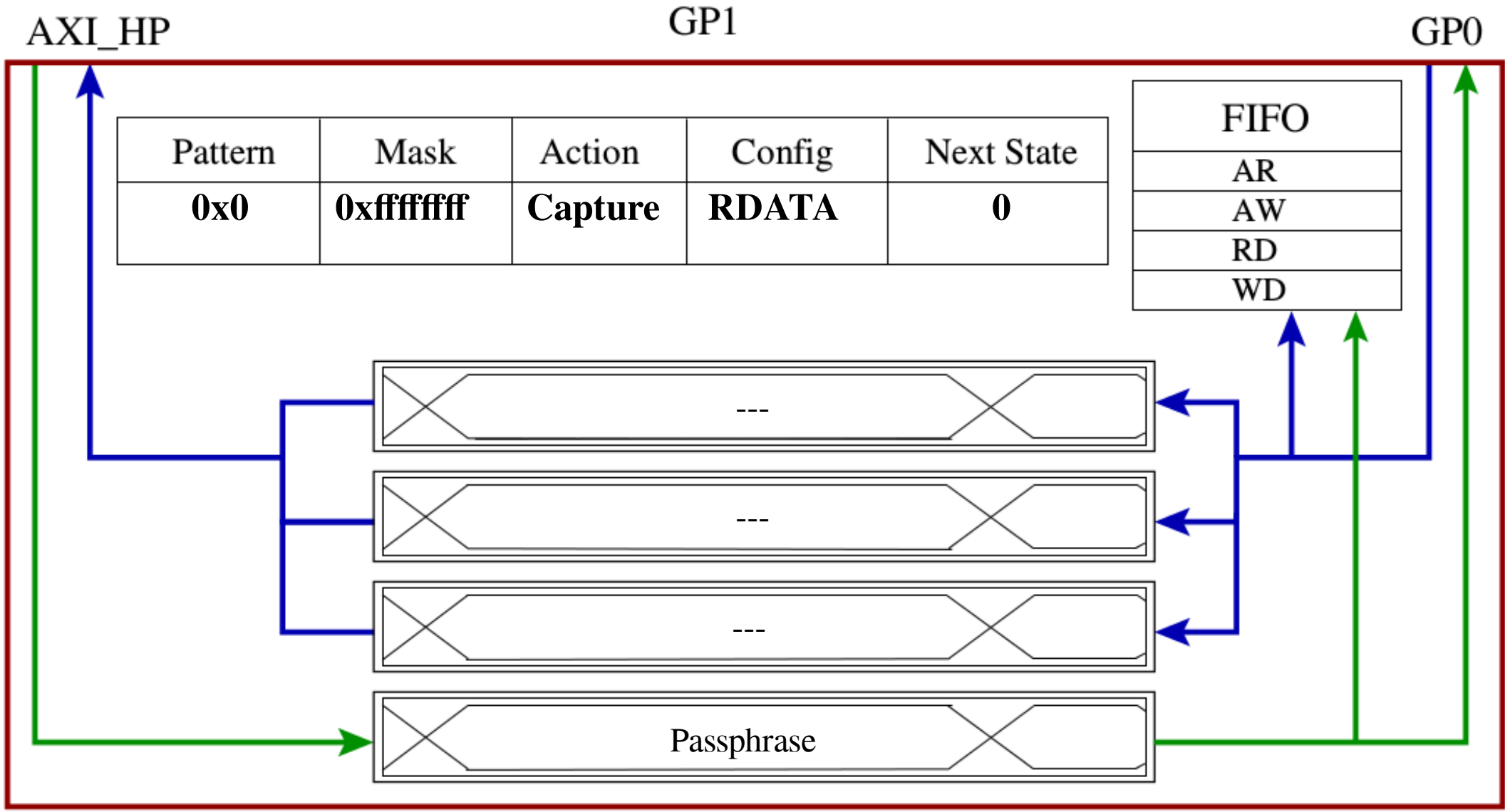
0x1F00AAAA


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
Instr 1
0xffffffff
None
RDATA
2
FIFO
AR
AW
RD
WD

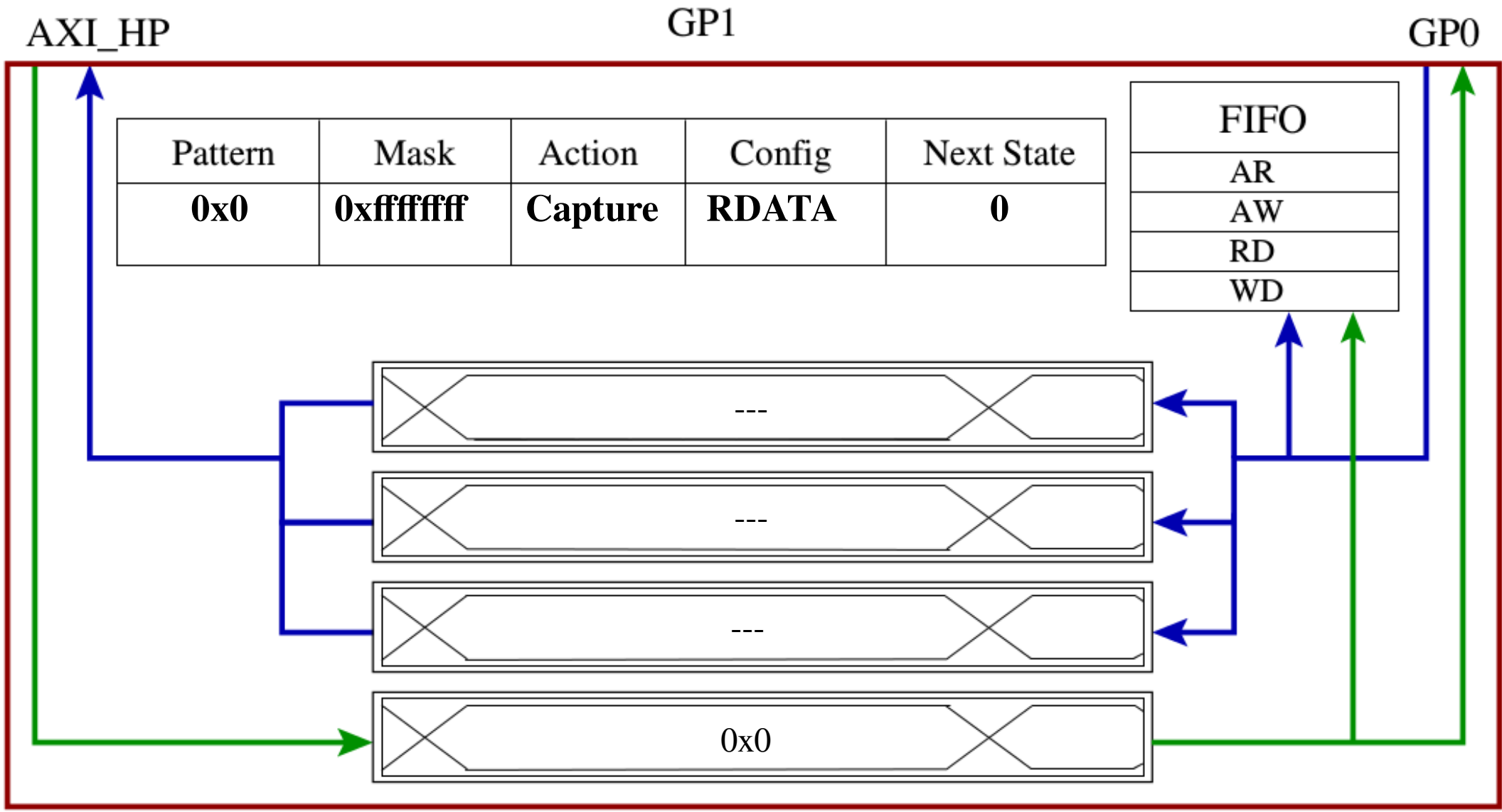
Instr1


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
Instr 2
0xffffffff
None
RDATA
3
FIFO
AR
AW
RD
WD

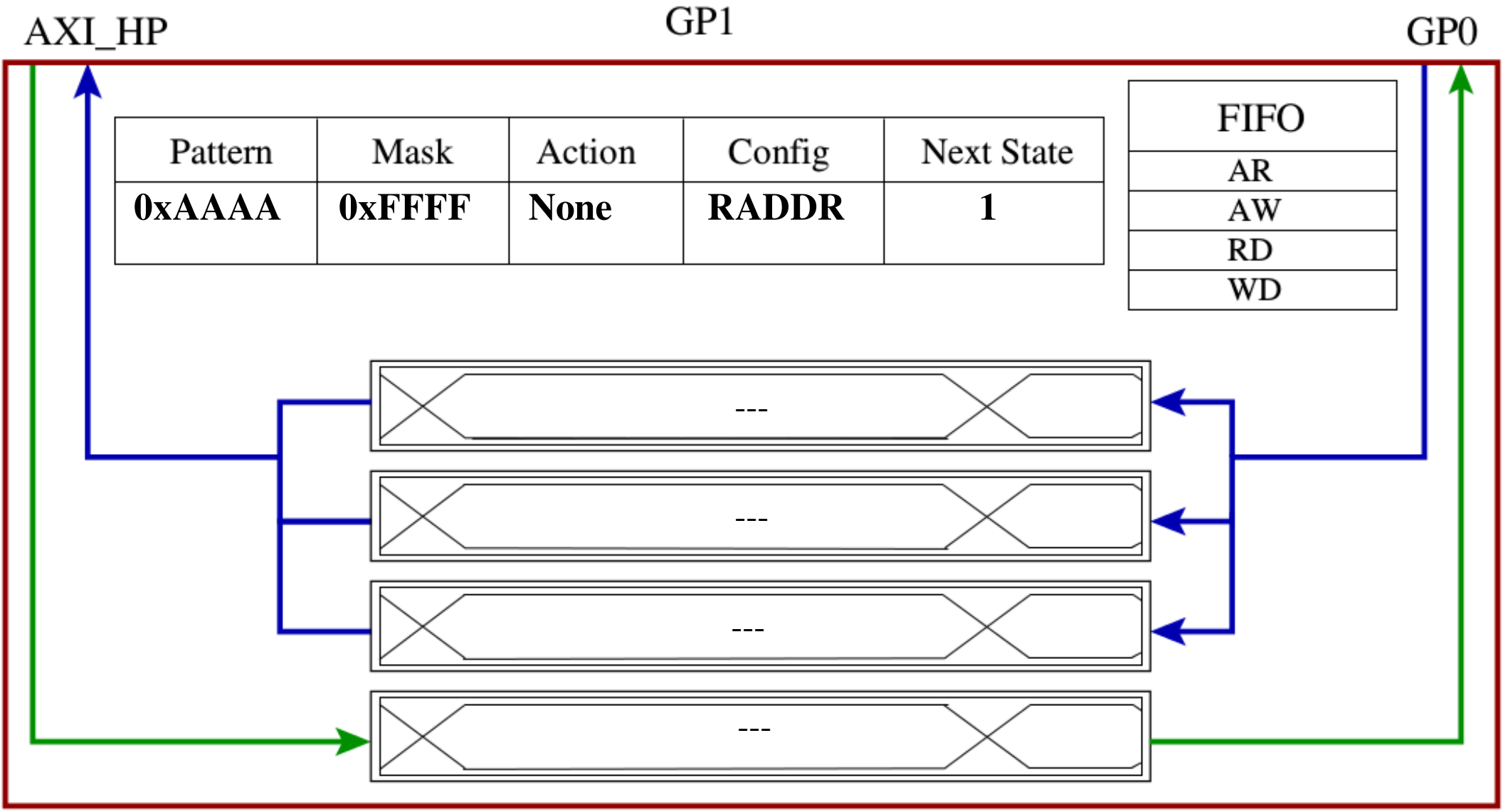
Instr2


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
0x0
0xffffffff
Capture
RDATA
0
FIFO
AR
AW
RD
WD

Passphrase


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
0x0
0xffffffff
Capture
RDATA
0
FIFO
AR
AW
RD
WD

0x0


AXI_HP
GP1
GP0
Pattern
Mask
Action
Config
Next State
0xAAAA
0xFFFF
None
RADDR
1
FIFO
AR
AW
RD
WD

Conclusion

- Memory tracer in FPGA
- All memory accesses visible
- Vulnerability of unsecured bus

Download our work!

https://secbus.telecom-paristech.fr/wiki/AxiBridge

Questions?