Hardware Complexity of Store-and-Forward, Cut-Through, and Preemption

Wolfram Lemppenau

University of Applied Science of Bocholt
Germany

email: wle@bocholt.fh-ge.de
phone: +49-2871-2155 814

Objective:

- **Better understanding how**

  store and forward - cut through - preemption

  influences hardware complexity

Strategy

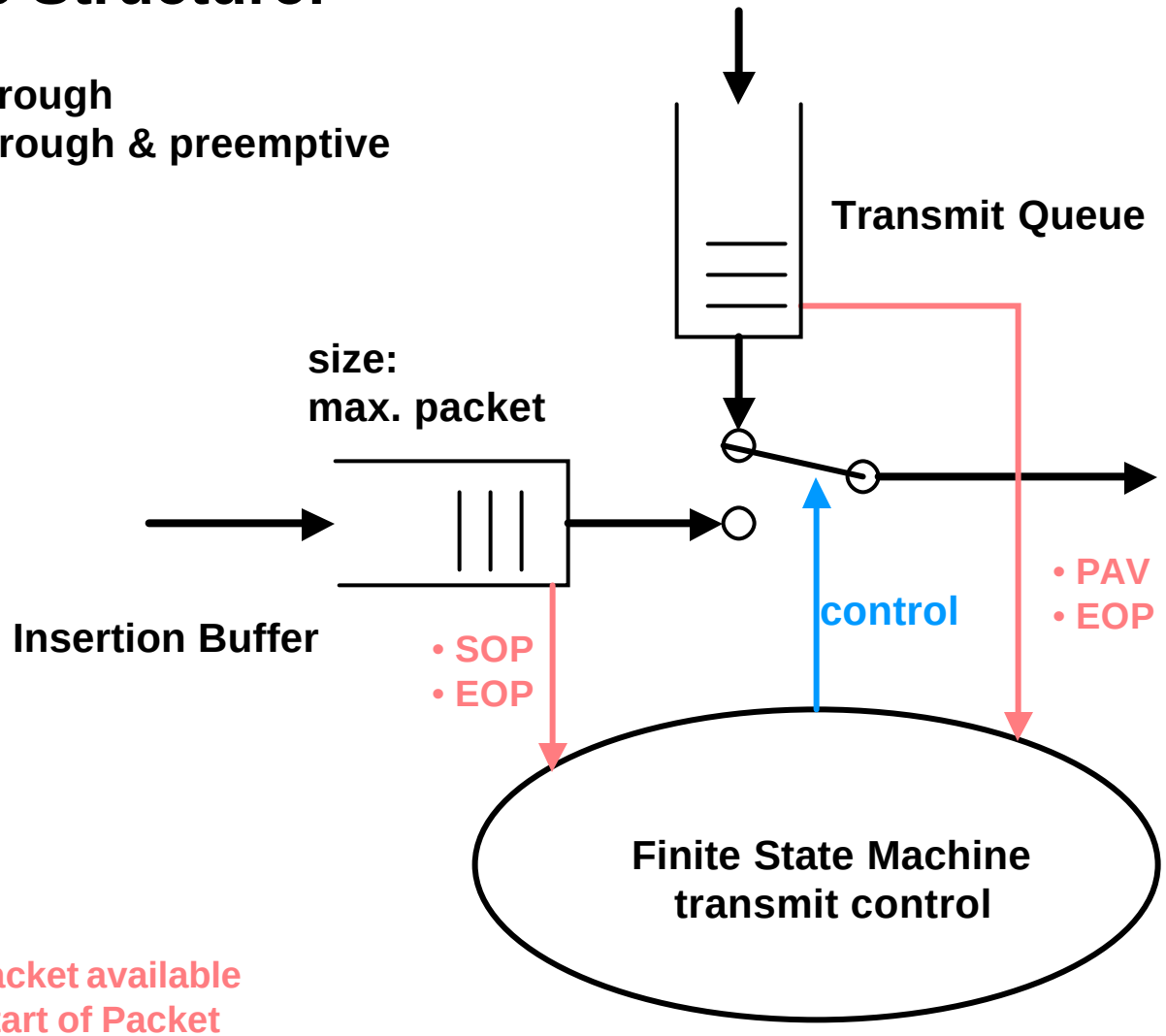
- **Examine hardware complexity based on real implementation**
- **Generic enough: parameters of interest dominate**
- **Detailed enough: realistic results**

It’s not

- **a product specific implementation**
- **MAC protocol dependent**

Node Structure:

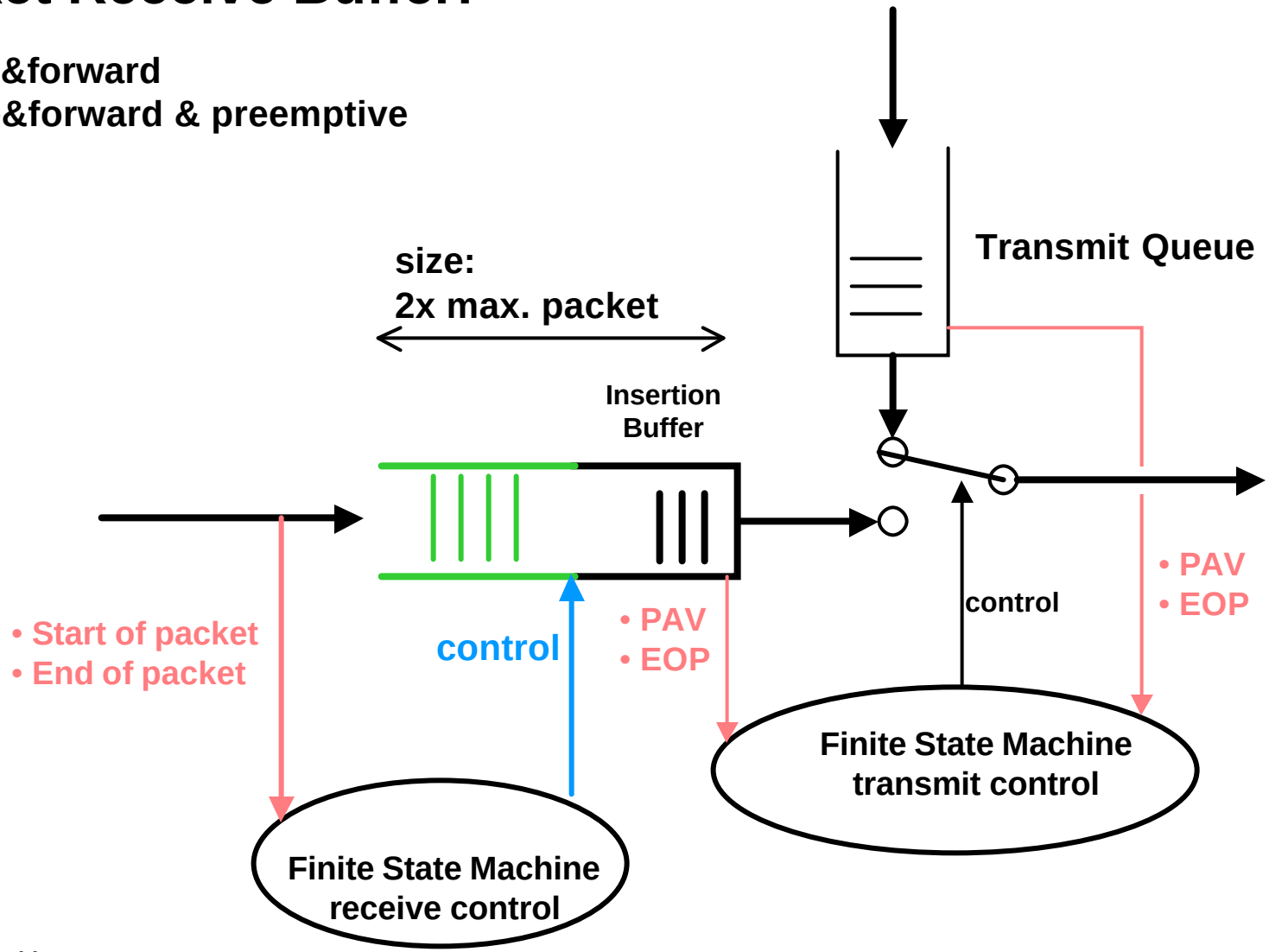
- cut through
- cut through & preemptive



- PAV: Packet available
- SOP: Start of Packet
- EOP: End of Packet

Packet Receive Buffer:

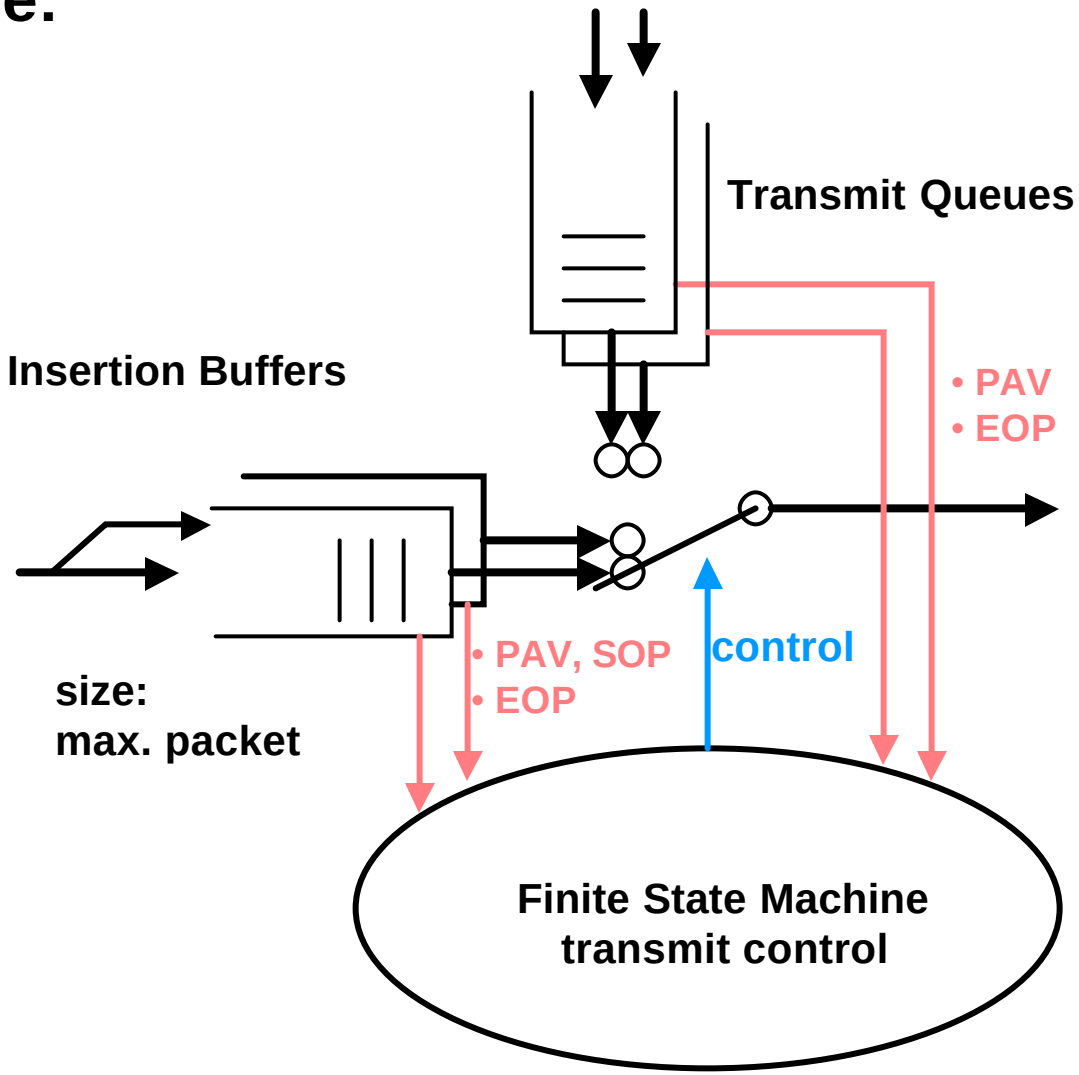
- store&forward
- store&forward & preemptive



FSM:
Finite State Machine

Node Structure:

•multiple priorities
(multiple queues)



FSM Implementation Details:

- **Type 1 / 2 difference: only different FSM implementation**
  global behavior is the same

- **All event combinations per state decoded**

- **Robustness: all unexpected (forbidden) events are treated as error,**
  error-state entry enforced
  (ignored otherwise, remain in actual state)

- **All unused “rest”-states (because of used state-encoding) enforce an error-state entry with error indication**

- **Binary state encoding, no state encoding optimization**

- **Synthesized and mapped (located) to FPGA (Xilinx) and CPLD (Lattice) architectures**

- **Optimized for area unless otherwise noted**

Considered Case Studies:

Case	Type	Priorities	Store&forward	Cut through	Preemptive	Robustness
A1/3	1/2	1	X	--	--	--
A2/4	1/2	1	X	--	--	X
B1/3	1/2	1	--	X	--	--
B2/4	1/2	1	--	X	--	X
C1/3	1/2	2	P1,P2	--	--	--
C2/4	1/2	2	P1,P2	--	--	X
C5/7	1/2	2	P1,P2	--	P1	--
C6/8	1/2	2	P1,P2	--	P1	X
D1/3	1/2	2	--	P1,P2	--	--
D2/4	1/2	2	--	P1,P2	--	X
D5/7	1/2	2	--	P1,P2	P1	--
D6/8	1/2	2	--	P1,P2	P1	X
E1	1	2	P2	P1	P1	--
E2	1	2	P2	P1	P1	X

Considered Case Studies:

Case	Type	Priorities	Store&forward	Cut through	Preemptive	Robustness
F1	1	3	P1,P2,P3	--	--	--
F2	1	3	P1,P2,P3	--	--	X
G1	1	3	--	P1,P2,P3	--	--
G2	1	3	--	P1,P2,P3	--	X
H1	1	3	P1,P2,P3	--	P1	--
H2	1	3	P1,P2,P3	--	P1	X
J1	1	3	P2,P3	P1	--	--
J2	1	3	P2,P3	P1	--	X
K1	1	3	P2,P3	P1	P1	--
K2	1	3	P2,P3	P1	P1	X

P1: Priority 1 (highest)

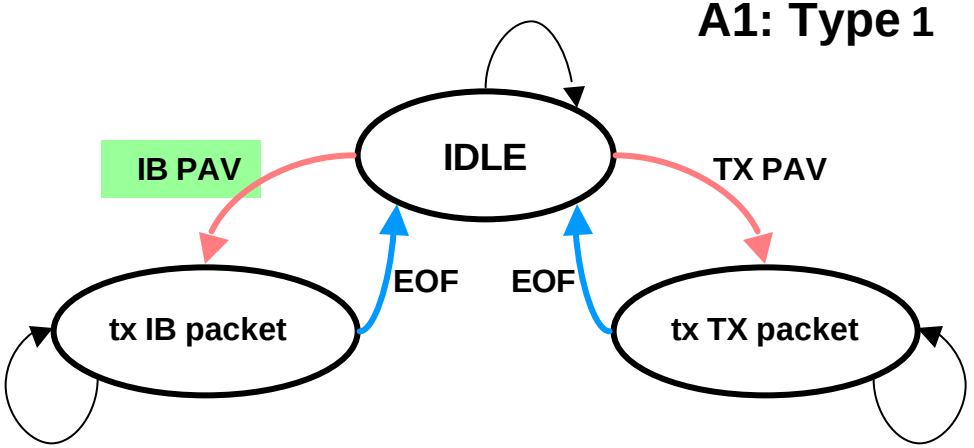
P2: Priority 2

P3: Priority 3 (lowest)

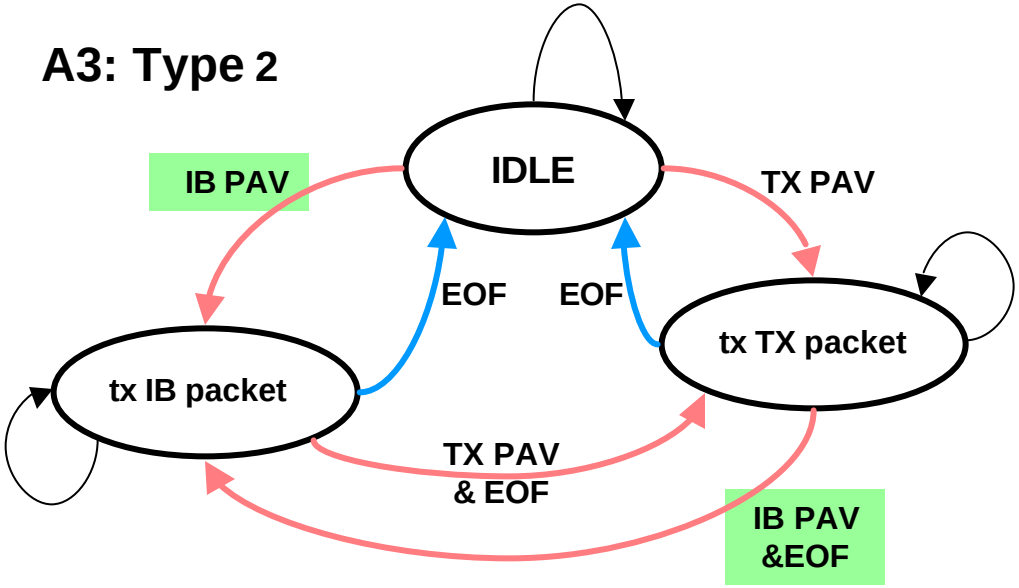
Case Studies A1, A3:

- one priority
- store&forward (non-preemptive)

A1: Type 1



A3: Type 2

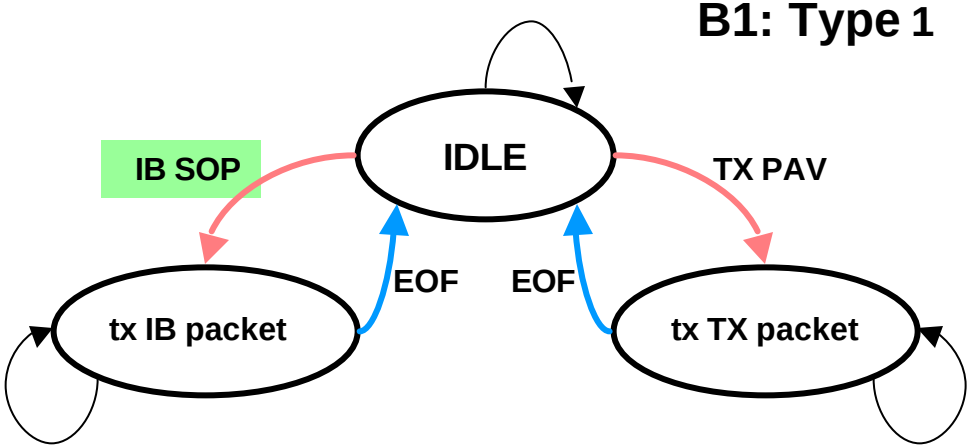


PAV: packet available
EOF: end of packet
TX: transmit buffer
IB: insertion buffer

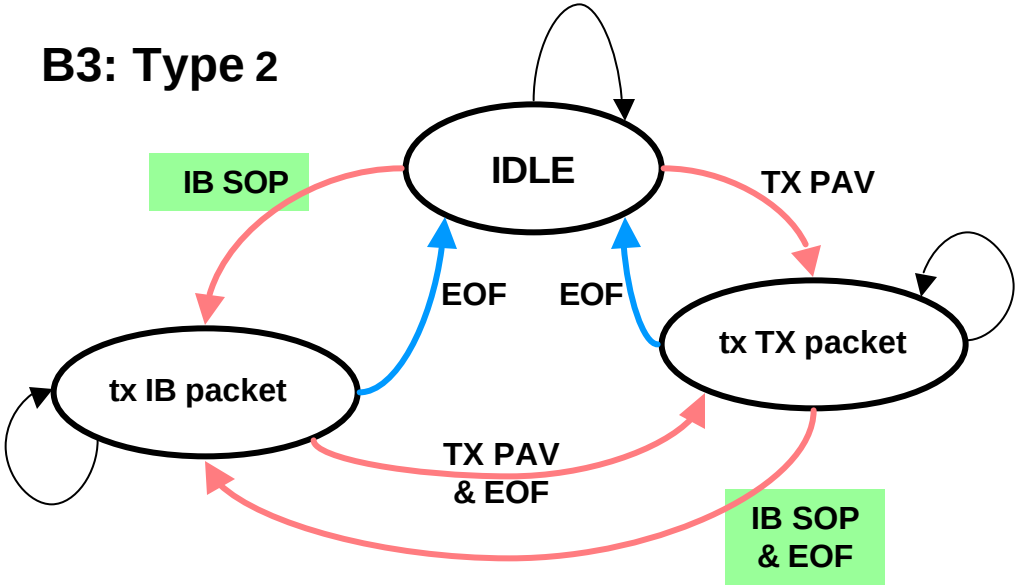
Case Studies B1, B3:

- one priority
- cut through (non-preemptive)

B1: Type 1



B3: Type 2

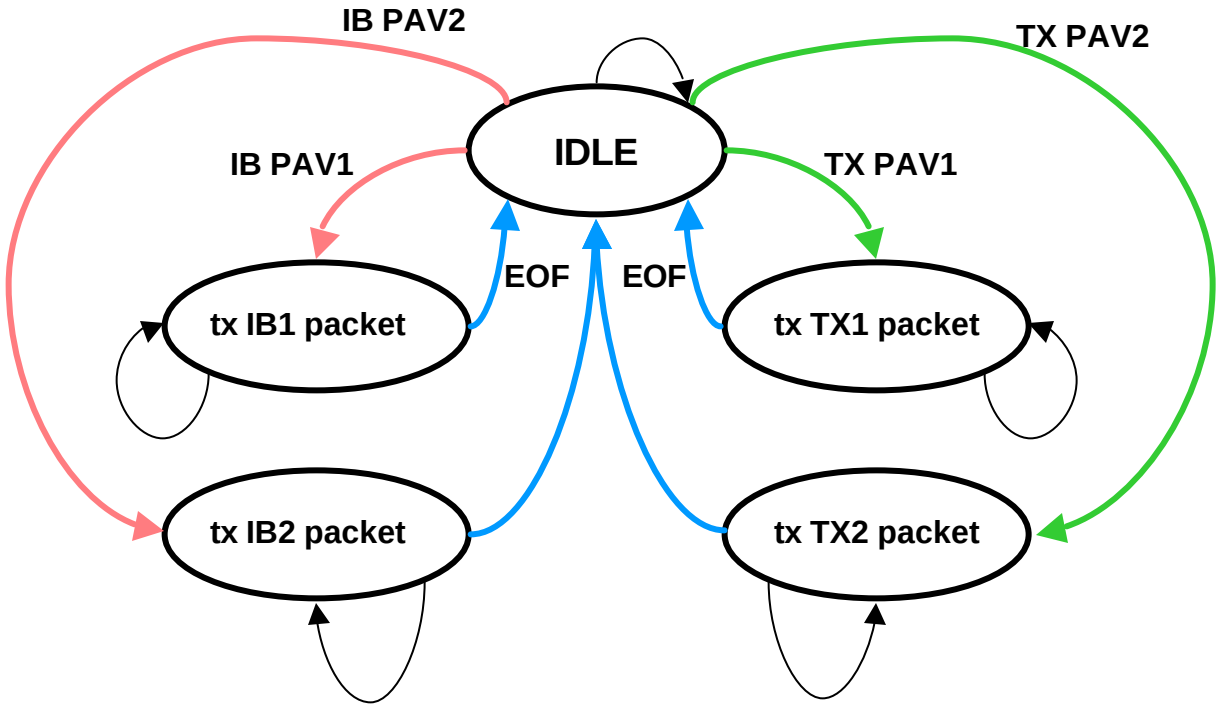


SOF: start of packet
EOF: end of packet
TX: transmit buffer
IB: insertion buffer
tx: transmit

Case Studies C1, C3:

- two priorities
- store&forward
- non-preemptive

C1: Type 1



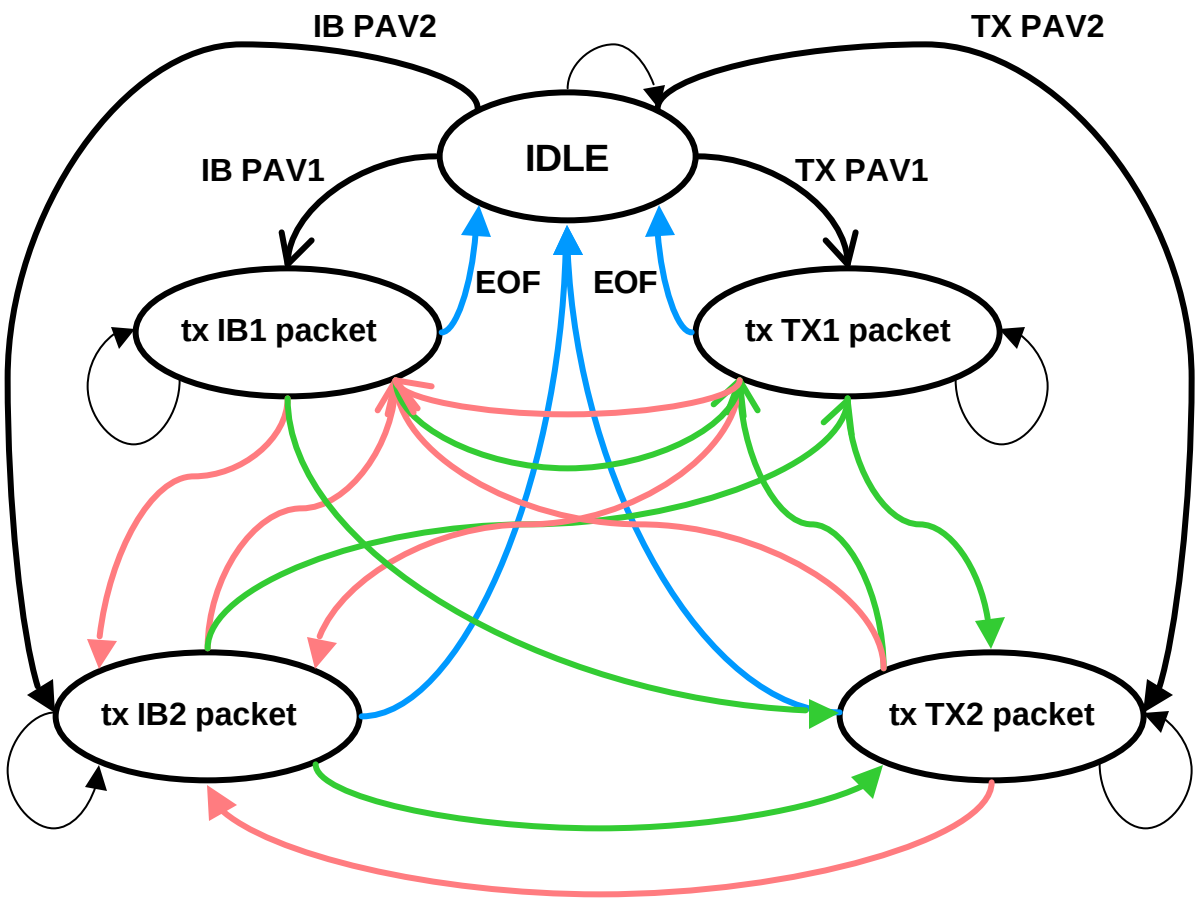
PAV: start of packet
EOF: end of packet
TX: transmit buffer
IB: insertion buffer

...1: priority 1
...2: priority 2
tx: transmit

Case Studies C1, C3:

C3: Type 2

- two priorities
- store&forward
- non-preemptive

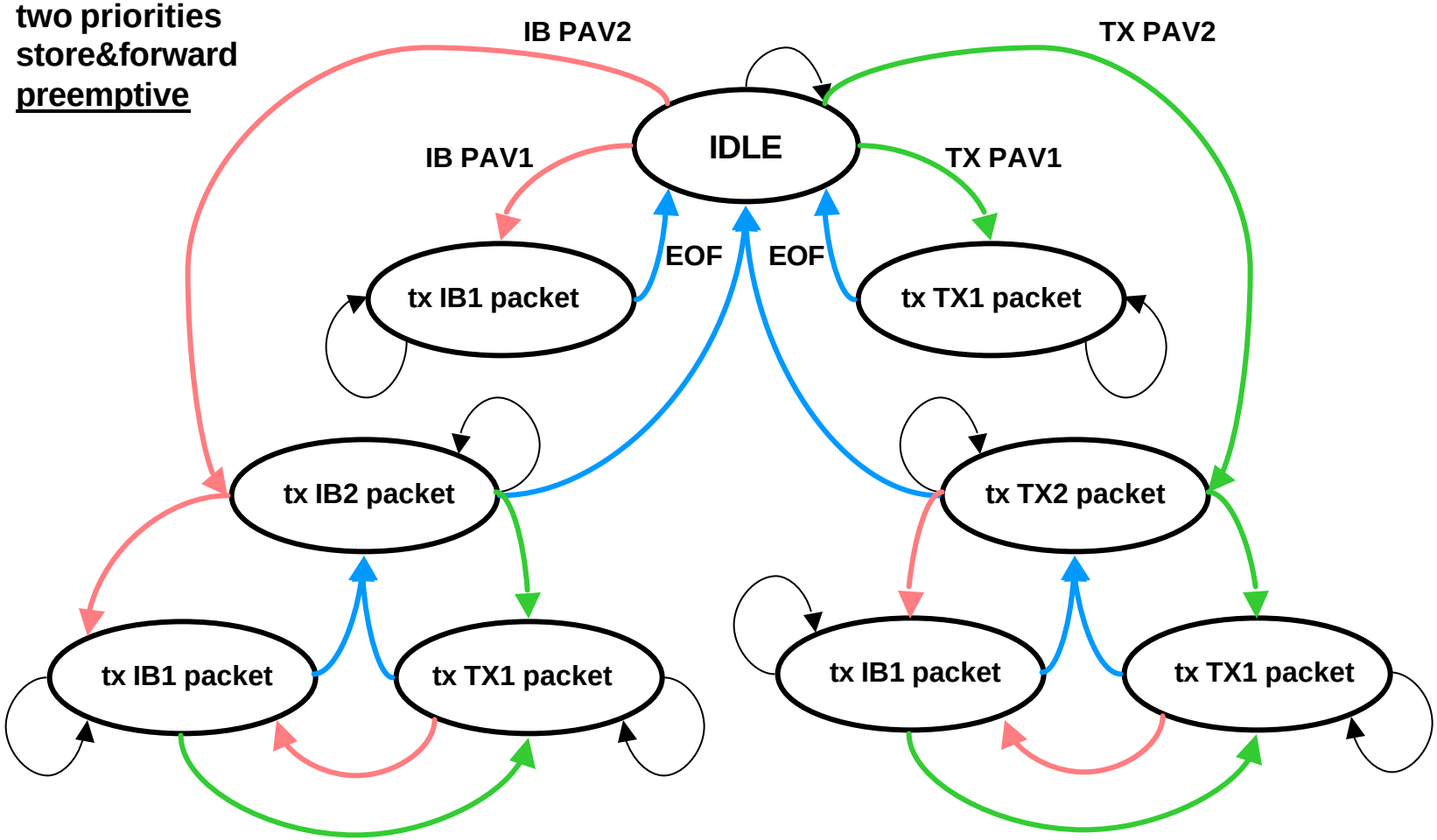


C1, C3:
same behavior in controlling outgoing packets

Case Study C5:

C5: Type 1

- two priorities
- store&forward
- preemptive



- D5: two priorities, cut through, preemptive
  almost identical

Results (1):

Case	Type	Priorities	S&f	Cut through	Preemptive	Robustness	FPGA [LBs]
A1/3	1/2	1	X	--	--	--	4 / 6
A2/4	1/2	1	X	--	--	X	7 / 7
B1/3	1/2	1	--	X	--	--	4 / 6
B2/4	1/2	1	--	X	--	X	7 / 7
C1/3	1/2	2	P1,P2	--	--	--	11 / 14
C2/4	1/2	2	P1,P2	--	--	X	24 / 40
C5/7	1/2	2	P1,P2	--	P1	--	25 / 29
C6/8	1/2	2	P1,P2	--	P1	X	53 / 78
D1/3	1/2	2	--	P1,P2	--	--	13 / 22
D2/4	1/2	2	--	P1,P2	--	X	27 / 32
D5/7	1/2	2	--	P1,P2	P1	--	31 / 37
D6/8	1/2	2	--	P1,P2	P1	X	62 / 87
E1	1	2	P2	P1	P1	--	31 (37)
E2	1	2	P2	P1	P1	X	56 (65)

LB: Logic Blocks

number in brackets: 1-hot encoding

Results (2):

Case	Type	Prio.	S&f	Cut through	Preemptive	Robustness	FPGA [LBs]
F1	1	3	P1,P2,P3	--	--	--	16 (25)
F2	1	3	P1,P2,P3	--	--	X	36 (43)
G1	1	3	--	P1,P2,P3	--	--	16 (25)
G2	1	3	--	P1,P2,P3	--	X	36 (43)
H1	1	3	P1,P2,P3	--	P1	--	46 (60)
H2	1	3	P1,P2,P3	--	P1	X	95 (116)
J1	1	3	P1,P2	P1	--	--	16 (43)
J2	1	3	P1,P2	P1	--	X	36 (43)
K1	1	3	P1,P2	P1	P1	--	46 (60)
K2	1	3	P1,P2	P1	P1	X	95 (116)

number in brackets: 1-hot encoding

Interpretation (1):

- **What impact has state encoding (look at e.g. E1...K2)**

  binary/1-hot encoding: minor difference **(FPGA: 1-hot encoding even worth)**

- **What impact has FSM robustness taken into account (look at e.g. D1/D3, K1/K2)**

  FSM robustness requires 100% more resources

- **How much extra resources required by type 2 ? (look at e.g. C1/C2, D6/D8)**

  type 2 requires 100% / 30% more resources

Interpretation (2):

- **Compared to store&forward, how much extra resources required for cut through ?**
  (look at e.g. F1/G1, F2/G2)

  cut through requires no additional resources **(no packet receive buffer)**

- **Compared to non-preemptive, how much extra resources required for preemptive ?**
  (look at e.g. F1/H1, J2/K2)

  preemptive requires 3x of resources **(BUT: consider absolute values)**

- **Regarding preemptive: are the absolute required resources design-dominant ?**

  NO, consider required resources e.g. for a SPI-4(2) interface

  RPR: ~100 logic blocks **SPI-4(2): ~4000 logic blocks**

SPI-4(2): System Packet Interface Level 4 Phase 2: OC-192 System Interface for Physical and Link Layer Devices
OIF: Optical Internetworking Forum