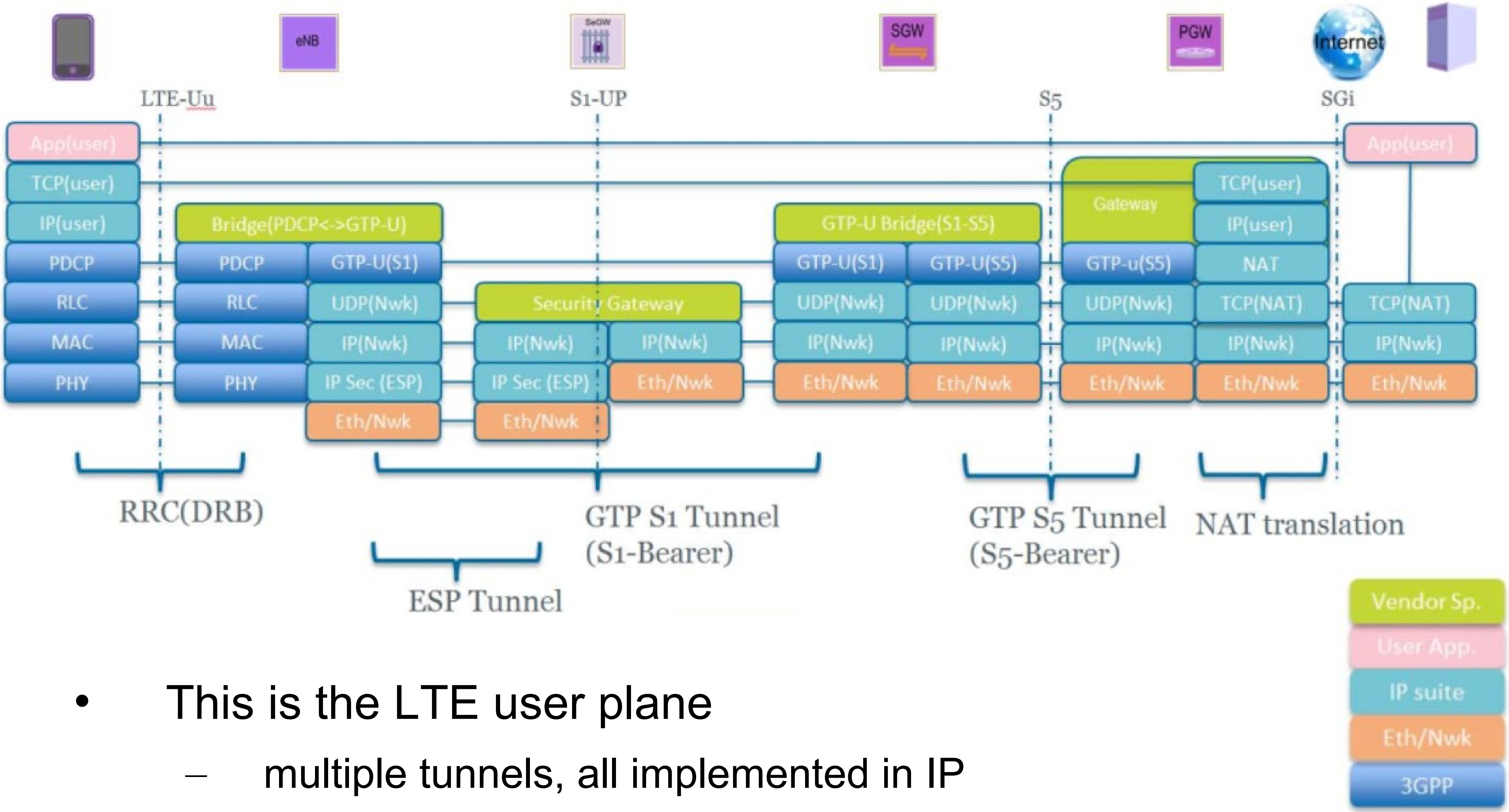
ETSI work on Next Generation Protocols for 5G

- John Grant, Nine Tiles
- Chairman, ETSI ISG NGP

ETSI ISG NGP

- Industry Specification Group, Next Generation Protocols
- First meeting in January 2016 at BSI (Chiswick)
  - initiative mainly from UK, including 5GIC (located at University of Surrey)
- Phase 1 of 5G (3GPP Release 15) concentrates on New Radio
  - more bits/s/Hz as well as new wavebands
  - lower latency
- ISG NGP is working on core and access networks, targeting Release 17
  - fix operators' problems with IP-based LTE core
    - natively support mobility, security, etc
  - more application bits/s/Hz
  - support new services proposed for 5G, e.g. sub-ms latency
  - new transport protocols
    - TCP mistakes delays on the air interface for congestion



- This is the LTE user plane
  - multiple tunnels, all implemented in IP

NGP work items

- Completed
  - GS 1: Scenario definitions
  - GS 2: Self-organising control and management planes
  - GR 3: Packet routing technologies
  - GR 4: Identity oriented networks
  - GS 5: Requirements
  - GR 6: Intelligence-defined network
  - GS 7: Reference model
- In progress
  - GR 8: Mobile deterministic networking
  - GS 9: Architecture
  - GR 10: New transport technologies
  - GR 11: Slicing
  - GS 12: Key performance indicators
  - GS 13: User plane packet formats and forwarding mechanisms

The rest of this talk outlines GS12 and then describes the two most radical features of GS13

Operators’ requirements

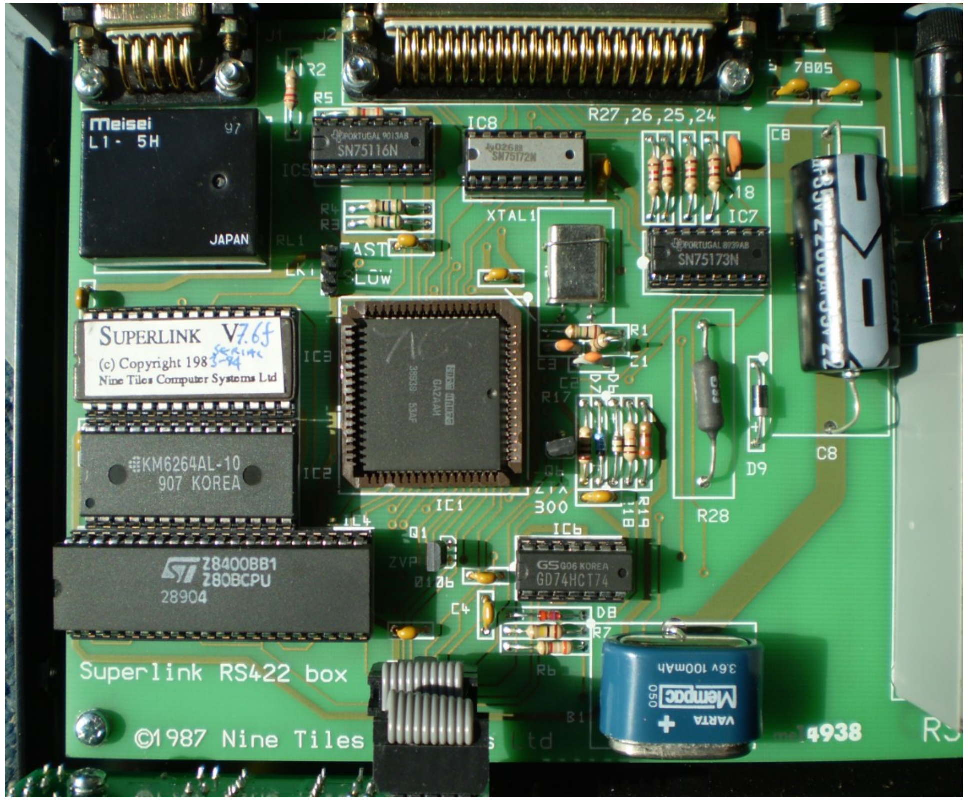
Operators' requirements: headline KPIs

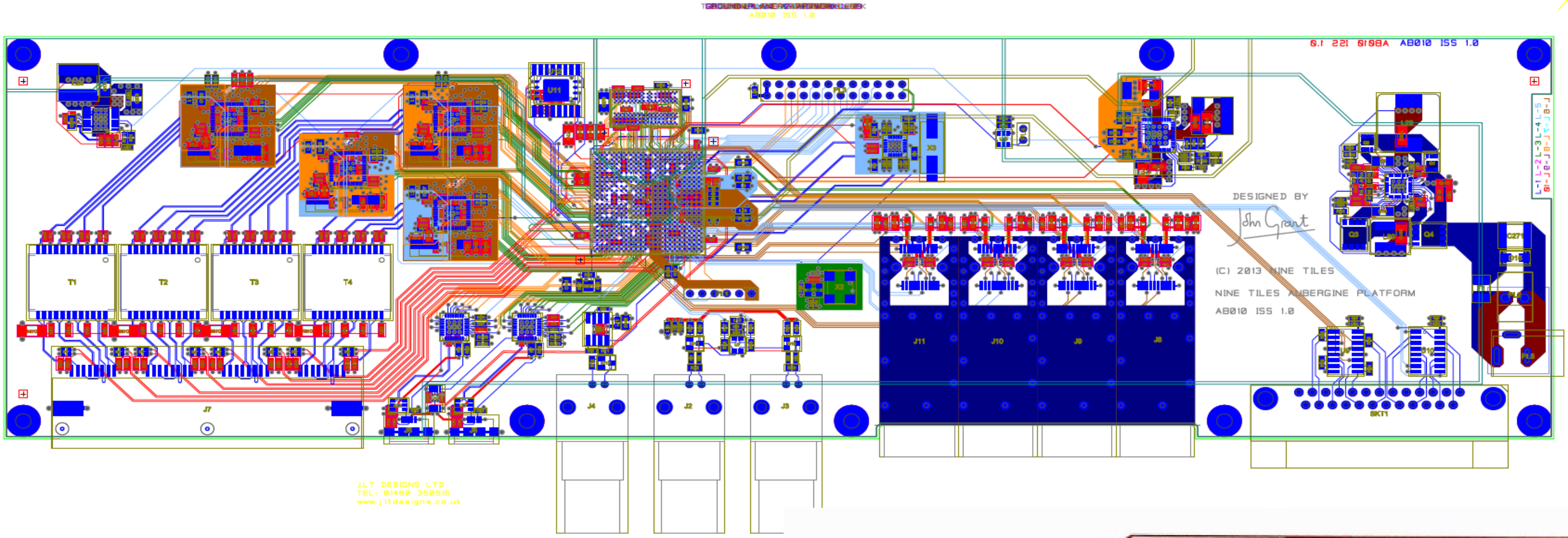
- Addressing
  - identify the entity, not the interface
    - identity must not be conflated with location
- Efficiency
  - minimise the size of packet headers and the amount of processing they need
    - IP header compression (used for voice over LTE) is power-hungry
- Latency
  - meet requirements for URLLC
  - new transport protocols that aren't slowed by retransmissions on the air interface
- Security (built in, not an add-on)
  - resist DDoS; avoid attack vectors such as "well-known ports"
    - operators are using NAT with IPv6 to avoid security risks of fixed addresses
- Interworking with previous generations and with the Internet

Evolution of digital platforms

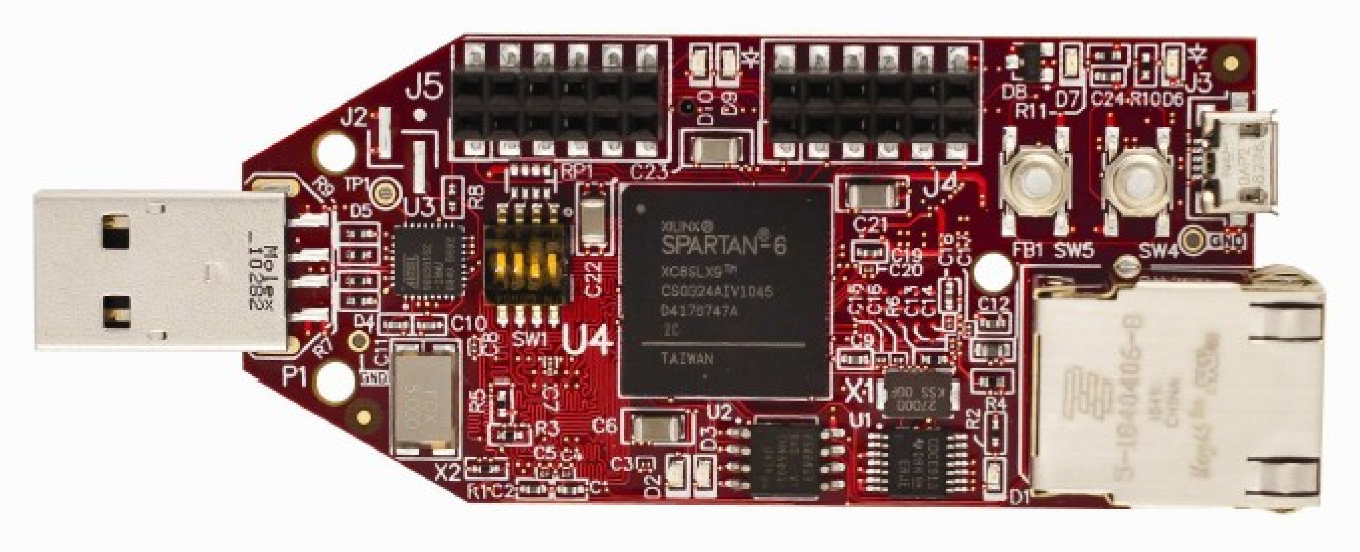
- 1970s
  - Arpanet IMP
  - almost everything in software
    - hardware calculation of checksum
      - maybe to reduce memory accesses by the CPU
  - limited memory (64KB address space)
    - connectionless routing
      - to avoid keeping state
      - everything needed for routing must be in the header
      - still needs state to route global addresses
  - line speed 0.056 Mb/s
  - 0.6 Mb/s (per pin) memory interface

- 1980s
  - Nine Tiles Superlink
  - almost everything in software
    - ASIC: few hundred gates
  - limited memory (64KB address space)
    - connection-oriented routing
      - to reduce per-packet processing
      - can connect by name or location
      - only need flow state, not routing information
  - line speed 1.5 Mb/s
  - 2 Mb/s memory interface

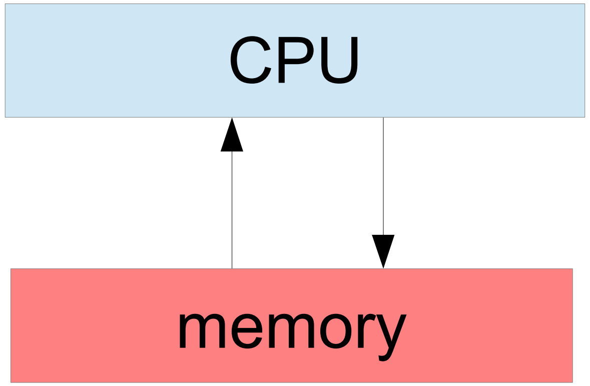




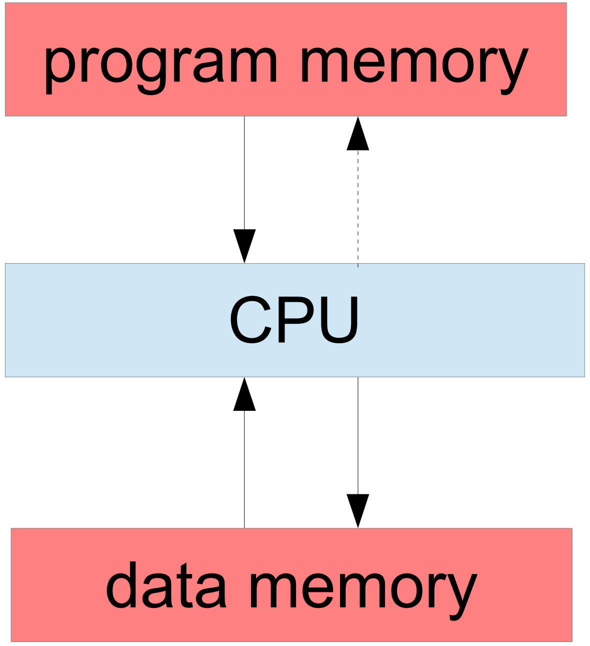
- 2010s
  - can do much more in logic
  - multiple Gb/s line speed
  - few Gb/s memory interface
  - dRAM chip sizes in Gb



- Computing systems are sequential
  - with everything passing over the memory interface
    - incoming data must be buffered until the CPU is ready to look at it
  - and memory speeds haven't increased as much as other parameters

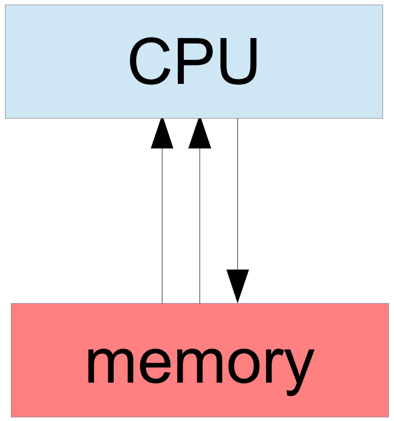


von Neumann architecture



Harvard architecture

code: a batch process



more complexity → takes more time

more data → takes more space

locate data by memory address (random access)

processes compete for CPU time and memory bandwidth

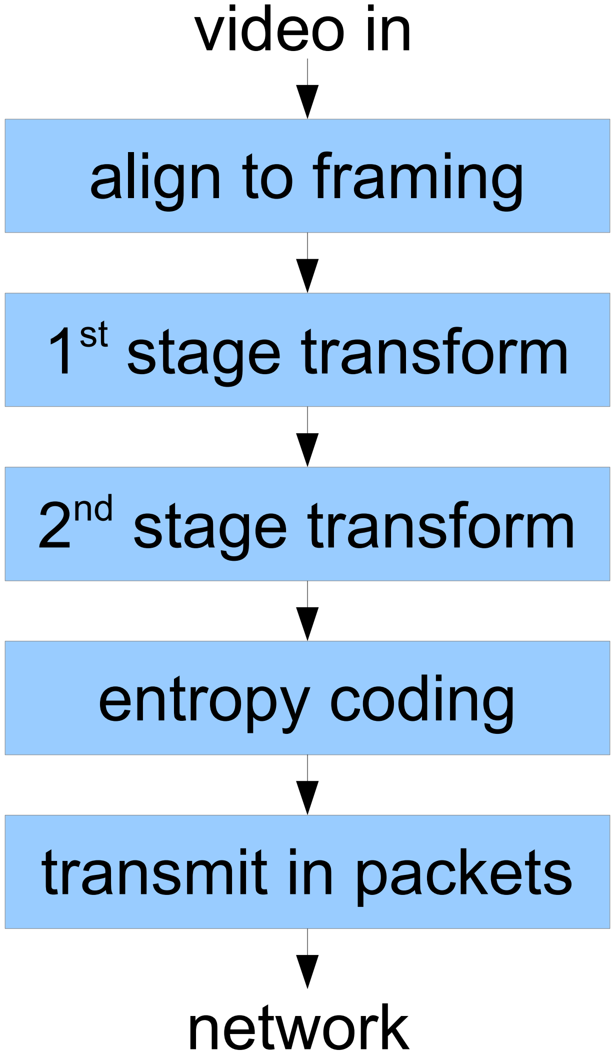
logic: a continuous process

more complexity → takes more space

more data → takes more time

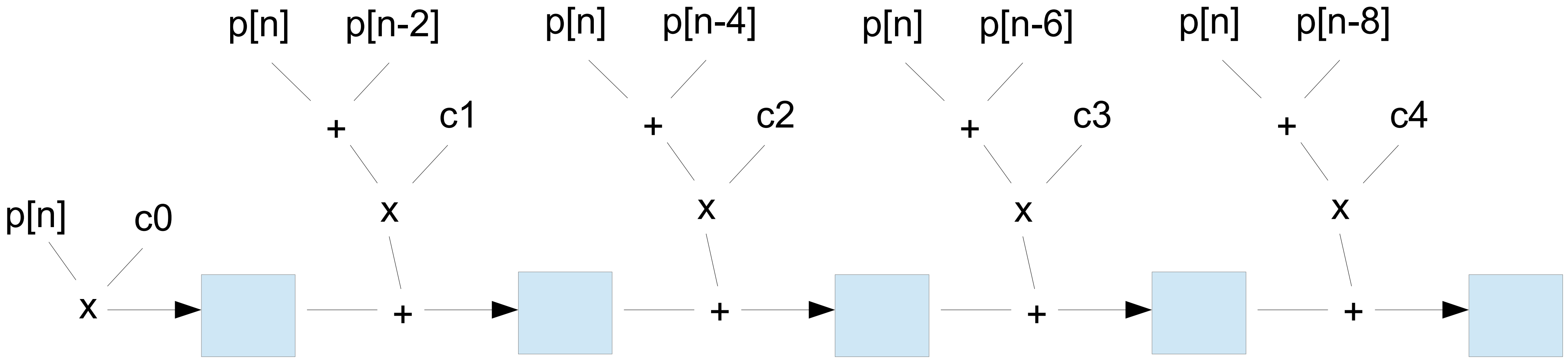
locate data by time of arrival (sequential)

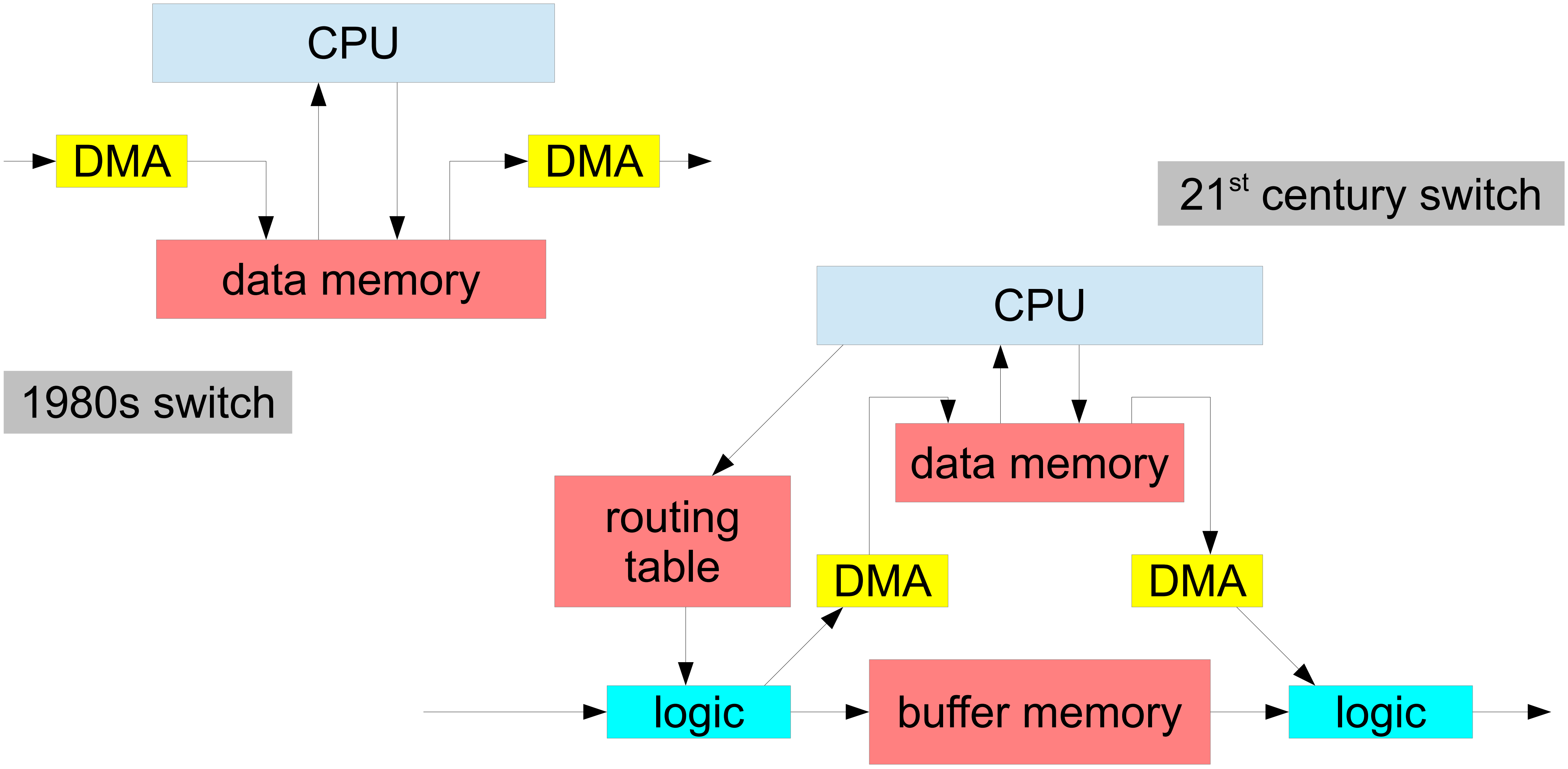
processes run independently



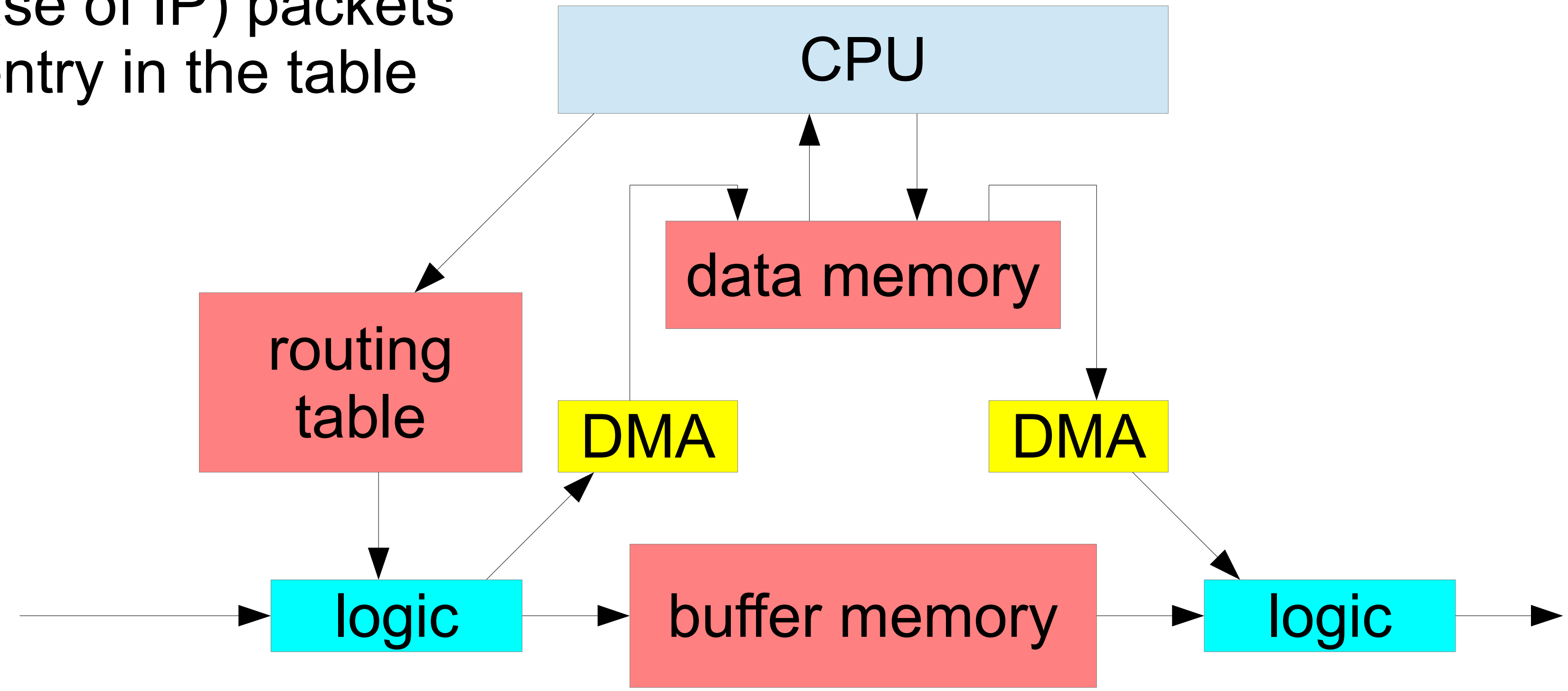
Example: wavelet transform

- 9 pixel values and 5 coefficients per pixel (for each component)
    - c0p[n] + c1(p[n-1] + p[n+1]) + c2(p[n-2] + p[n+2]) + … + c4(p[n-4] + p[n+4])
  - in logic a 5-stage multiply-and-accumulate at pixel clock rate
    - 5-clock pipeline: pixel values used by each block in turn




CPU
DMA
DMA
data memory
1980s switch
21st century switch
CPU
data memory
routing table
DMA
DMA
logic
buffer memory
logic

- Most packets don't go through the CPU's memory
- Entry in the routing table shows how to route each “flow”
- Control and management packets routed to the CPU
  - also (in the case of IP) packets for which no entry in the table

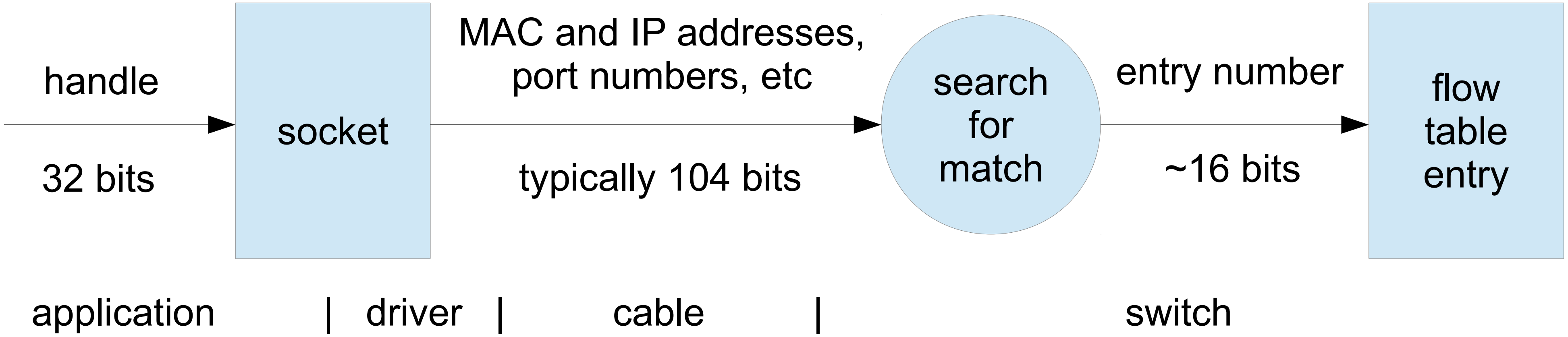


IP is connectionless to save memory, but ...

- Memory is no longer a scarce resource
  - sockets keep information about flows in the endsystems
  - SDN controllers keep information about flows
  - routing tables keep information about flows

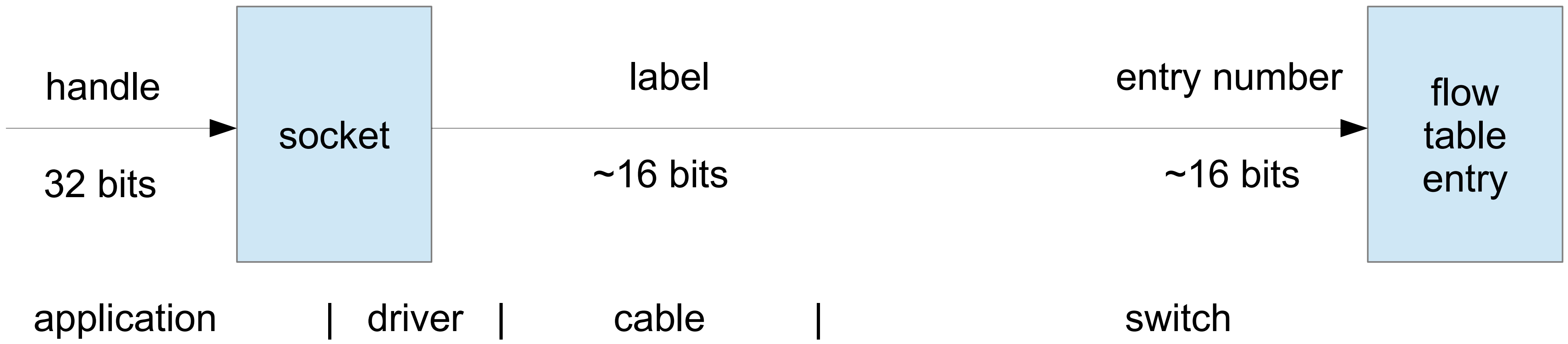
IP is connectionless to save memory, but ...

- Memory is no longer a scarce resource
  - sockets keep information about flows in the endsystems
  - SDN controllers keep information about flows
  - routing tables keep information about flows
  - how are those flows identified?



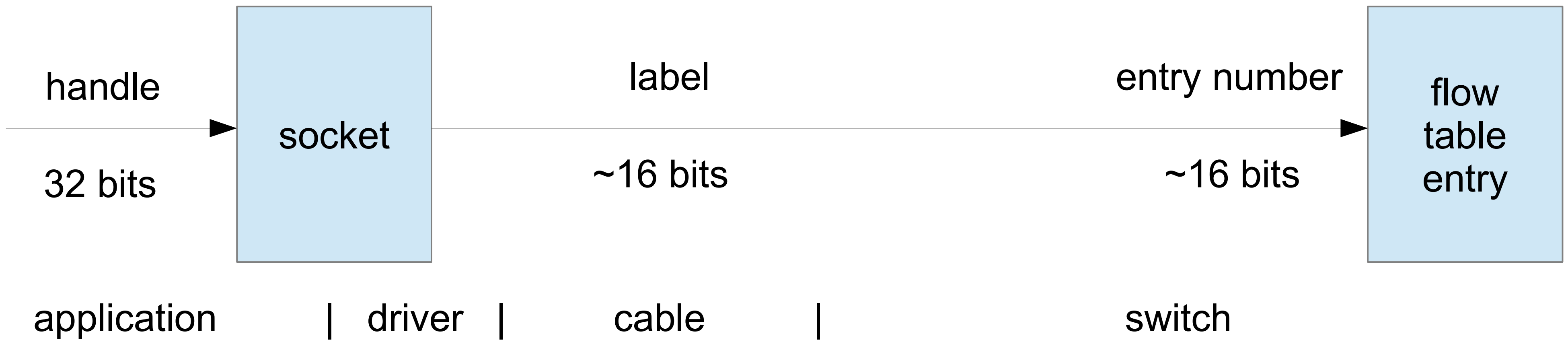
Proposed new protocol

- Header contains only the label and the length
  - all other information is in routing tables (control plane or data plane)
  - header format is local to the link
- Set route up on **`connect()`** instead of when first packet sent



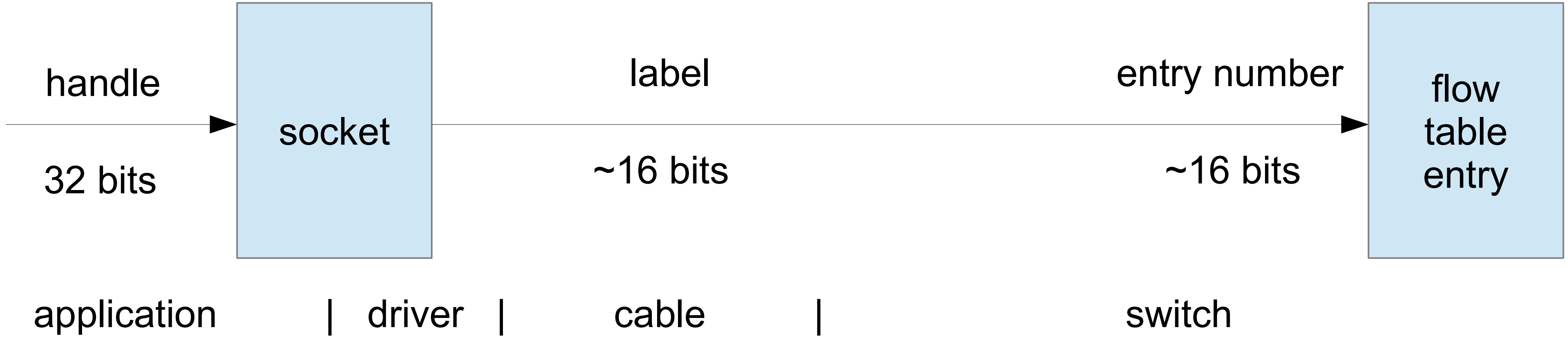
Proposed new protocol

- Can support much more communication with the application
  - including multiple addressing schemes
    - IPv4, IPv6, content-centric, service name, ...
    - maybe use domain name directly instead of translating to IP address
    - adding new address type does not require change to packet format



Proposed new protocol

- Can support much more communication with the application
  - also
    - QoS negotiation
    - security: authorisation, authentication, ...
    - format etc information for the remote entity



Steam age vs 21st century?

Live streams, “real world” signals

Time

- in IT (computing)
  - X must happen after Y
  - time per CPU instruction not well-defined
  - QoE depends on time to complete a process
  - few seconds latency is acceptable
- in AV (& other continuous media)
  - X is needed at time t
  - very precise word/pixel clocks
  - QoE depends on every sample arriving at the right time
  - 30ms delay mic-to-monitor impairs performance
  - 15ms motion-to-photon for VR
  - 1ms specified for tactile feedback

Packet networking as a best-effort service

- Competition for outgoing link in switches
  - can't predict offered traffic; requires queuing
  - longer queue → longer delay before forwarding
  - unbounded queue size + fixed memory size → packets can be dropped
- Good for unpredictable “IT traffic” (e.g. web surfing, file transfer)
  - using acknowledgement & retransmission as in TCP
- Not good for live continuous media where latency matters
  - including some of the new services proposed for 5G

Three levels of determinacy

also identified by
IETF DetNet group

- Best-effort
  - no guarantees

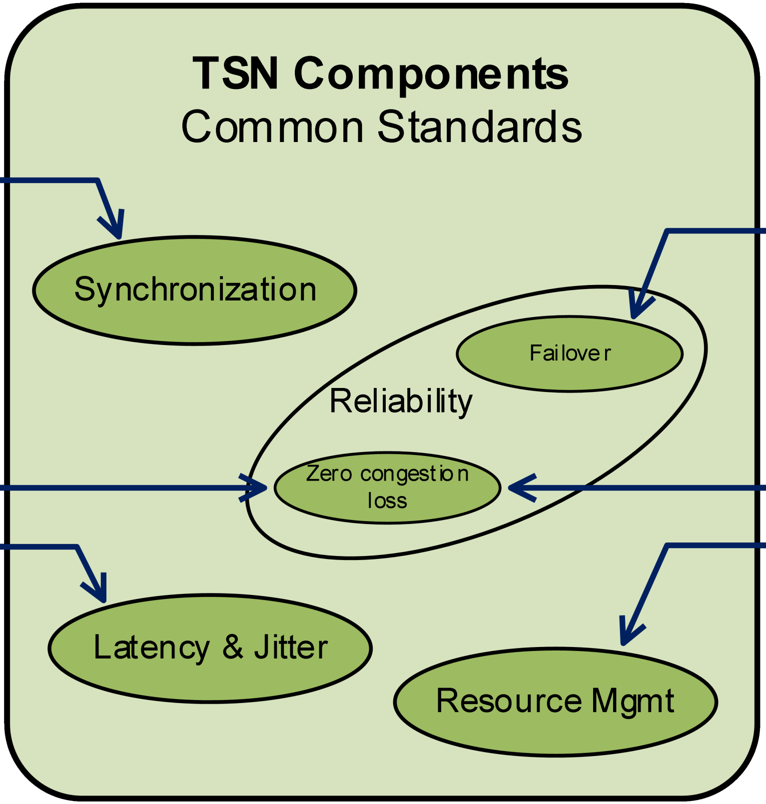
- Asynchronous
  - reserved capacity on each link
    - multiple queues with different priorities
  - bounded latency, no dropped packets

- Synchronous
  - scheduled transmission on each link
  - fixed latency, no dropped packets

Retrofitting timeliness to IT standards is complex

Time synchronization:
Timing and Synchronization (802.1AS)
Includes a profile of IEEE 1588

Bounded low latency and jitter:
Credit Based Shaper (802.1Qav)
Frame preemption (802.3br & 802.1Qbu)
Time-Aware Shaper (802.1Qbv)
Cyclic Queuing and Forwarding (802.1Qch)
Asynchronous Traffic Shaping (P802.1Qcr)



Failover:
Frame Replication and Elimination (P802.1CB)
Path Control and Reservation (802.1Qca)
Per-Stream Filtering and Policing (802.1Qci)
Reliability for time sync (P802.1AS-Rev)

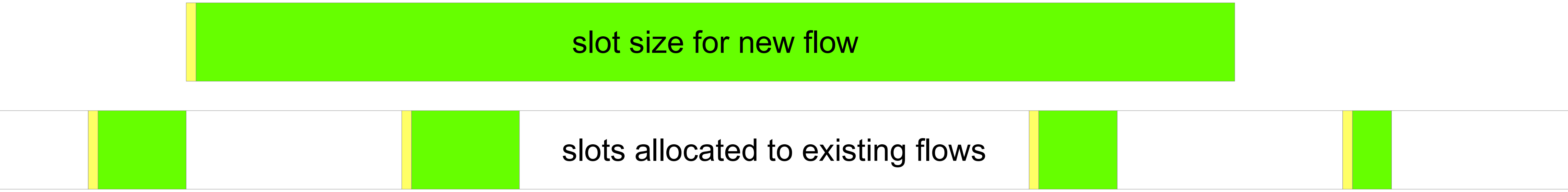
Dedicated resources & API
Stream Reservation Protocol (802.1Qat)
TSN configuration (P802.1Qcc)
YANG (P802.1Qcp)
Link-local Registration Protocol (P802.1CS)

NGP has a separate service for AV traffic

- wired links are formatted into “frames” divided into 64-byte “slots”
  - each slot can be allocated to a flow
    - carries a (variable-length) packet for that flow
    - latency is well-defined
    - per-flow allocation means no policing or shaping needed

NGP has a separate service for AV traffic

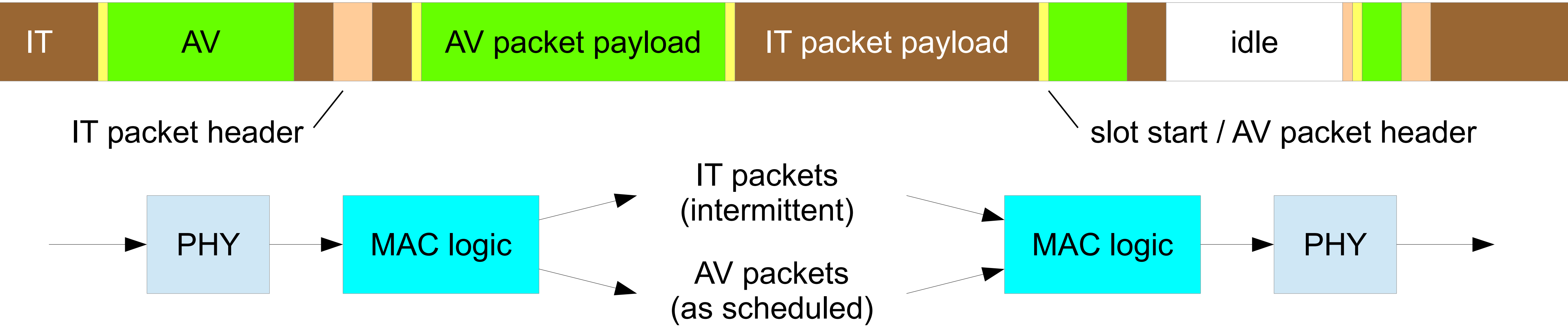
- wired links are formatted into “frames” divided into 64-byte “slots”
  - each slot can be allocated to a flow
    - carries a (variable-length) packet for that flow
  - originally intended to have variable-sized transmission slots
    - fixed-size found to be better in proof-of-technology implementation
      - difficult to allocate slots if some flows have frequent small packets and others have large packets
      - e.g. 96 kHz audio sample time 10.4 μs, 1500 bytes @ 1 Gb/s = 12 μs



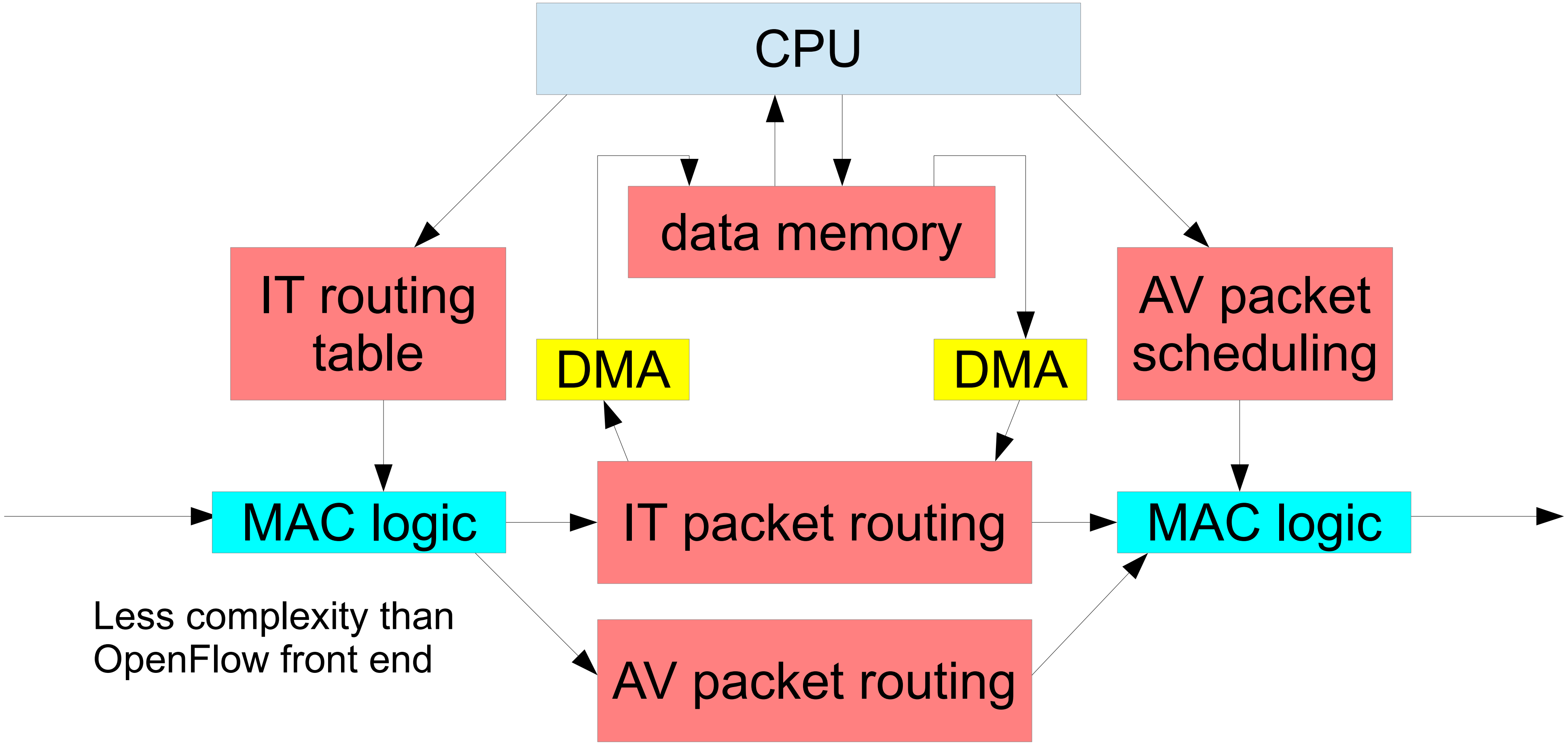
Synchronous service for AV traffic

- frames phase-locked on all links
  - very simple mechanism found to be effective
  - packets don't need labels
    - recipient can identify flow from time of arrival (or, equivalently, position in frame)
- all incoming data written to forwarding buffer
  - stays for a few microseconds until overwritten
  - schedule shows where to copy from for each slot
    - similar to cross-point audio and video switches
    - makes multicasting easy

- AV and IT services multiplexed together on wired links
  - can use a more sophisticated format with today's electronics
  - position AV (synchronous) packets where required ("foreground layer")
  - IT (best-effort) packets use remaining bytes ("background layer")
    - IT packets can be longer than a slot; no fragmentation headers needed



Switch structure



Less complexity than
OpenFlow front end

Control plane signalling

- IEC 62379-5-2 standard used in proof-of-technology implementation
  - TLV format, fixed part + Information Elements
  - 5G will probably use a different format

- FindRoute message
  - sets up flow
    - includes all per-flow information (which in IP is per-packet or via other protocols)
    - replaces DHCP, DNS, ARP, SIP, SDP, RSVP, ICMP, NAT, ...
  - similar process also required for connectionless technologies
    - less controllable, involves guesswork and folklore
      - e.g. assume UDP more urgent then TCP
    - “Reserving resources before packet transmission ... is impossible to avoid”
      - (4.3.2 in DetNet architecture draft)

- FindRoute message (continued)
  - wide variety of ways to identify the called party
    - equipment name etc in MIB
    - 64-bit unit identifier
    - Ethernet, IPv4, IPv6, E.164, etc, address
    - service name
    - content identifier
    - can prefix with a locator to define scope (recursively)
  - includes identification of data format, protocol, etc
    - supports negotiation between endpoints
    - also with the network, e.g. trade-off between image quality and bandwidth

- FindRoute message (continued)
  - can also include
    - QoS negotiation
    - security information (identification, authorisation)
    - service name (e.g. “Radio 4 LW”)
    - importance (e.g. on-air, “fader away from on-air”, just listening)
    - privilege level (listener/subscriber, operator, supervisor, maintenance)
    - charging information
    - ...

Security

- Routing only uses information from neighbour
  - signalling messages forwarded by neighbour after its own processing
    - if using SDN, area covered is a walled garden
  - can check before setting up flow in forwarding plane
    - more controllable than firewalls based on IP address and port number
    - message can include information on trustworthiness of upstream systems
    - possibility to notify “fingerprint” of DDoS attack to edge
  - forwarding plane only routes flows set up by control plane
    - IT flow label from neighbour’s routing table
    - AV slot positions assigned by neighbour
- Control plane messages get priority over other IT flows
  - can’t be impeded by excess of user traffic

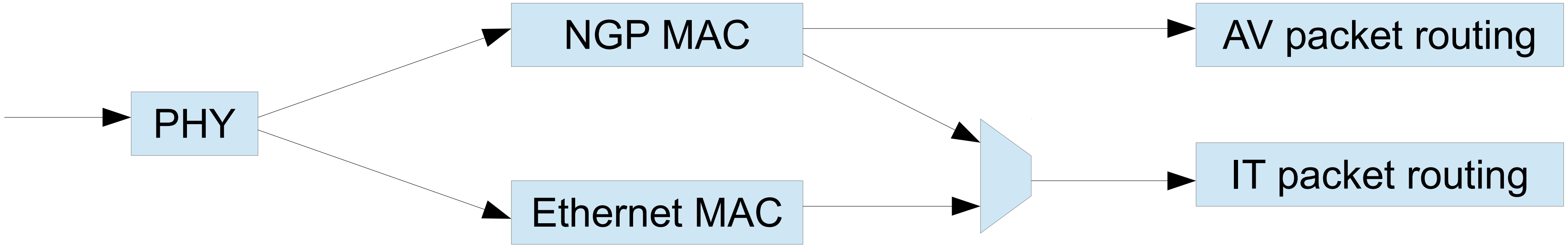
Transport layer options

- TCP
  - adjusts throughput to match network
  - retransmits dropped packets
  - provides an end-to-end data integrity check
- All still needed if used over the IT service
- For transport over the AV service
  - throughput can be negotiated between endpoints and network
  - no dropped packets
  - just need end-to-end data integrity check
  - confirmation of correct receipt can be combined with flow clear-down

Migration

- Legacy protocols over NGP
  - encapsulate in IT flows: similar to MPLS
  - can also implement a PseudoWire service using AV flows
  - bridge (include layer 2 header) or route (just send IP datagram)
- NGP over legacy protocols (“virtual link”)
  - encapsulation based on Audio Engineering Society AES51 standard
    - over Ethernet or UDP
    - service expected for AV flows is signalled in FindRoute message
- Enables gradual replacement of equipment
  - applications can use old or new protocols
  - small Flexilink islands can expand and coalesce
  - apps using new protocols see improved service when within an island

- Network ports auto-detect framing
  - current implementation supports both new and old
    - receive side auto-selects format



  - NGP MAC is selected if it recognises correct frames
  - Ethernet MAC is selected otherwise

Nine Tiles

John Grant

mailto: j@ninetiles.com

http://www.ninetiles.com/

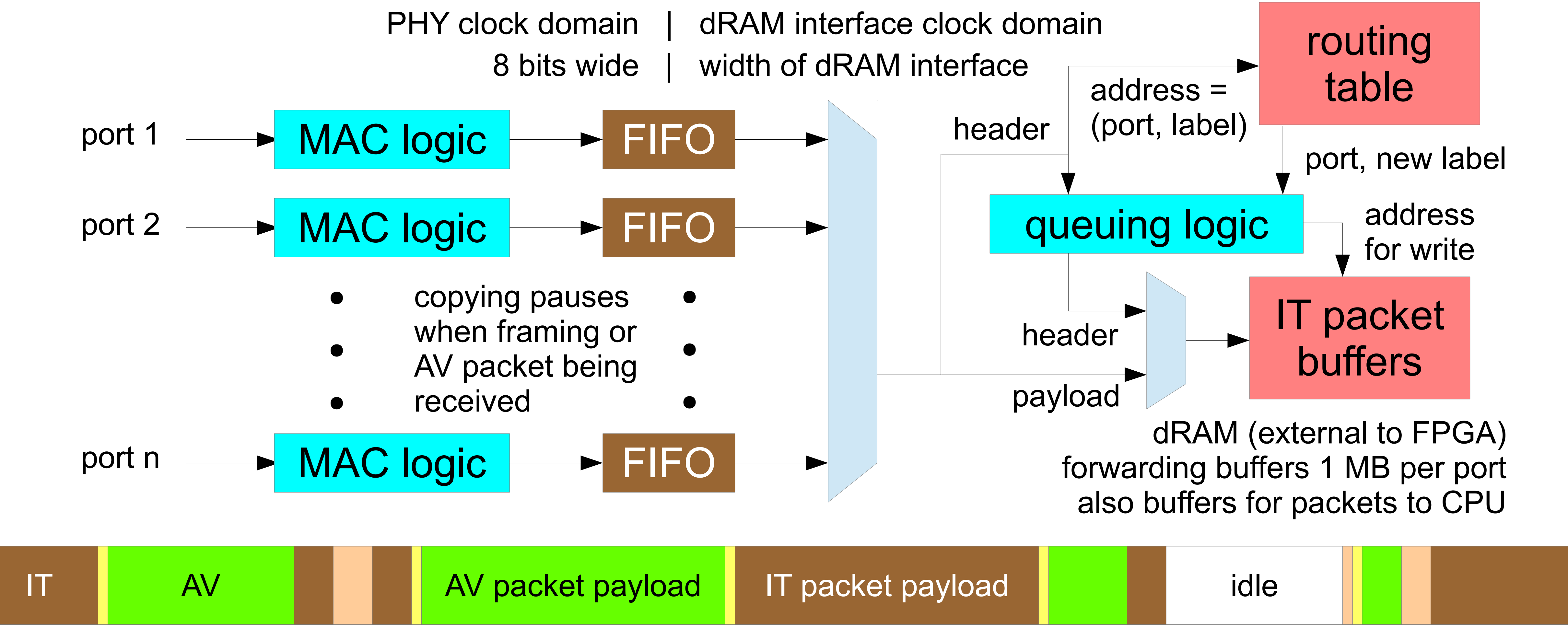
send e-mail to request participation in UK5G's open, informal group for development and trialling of NGP

ETSI ISG NGP

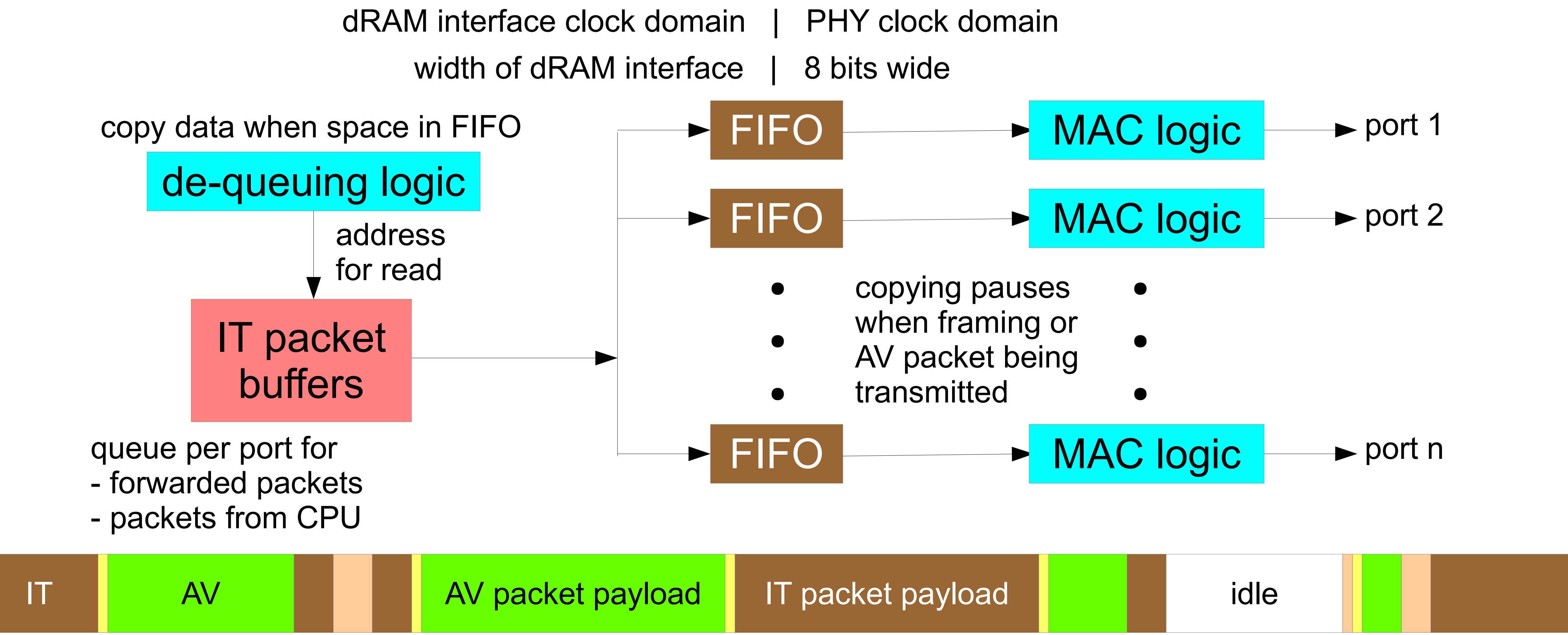
http://www.etsi.org/technologies-clusters/technologies/next-generation-protocols

Additional slides: packet routing details

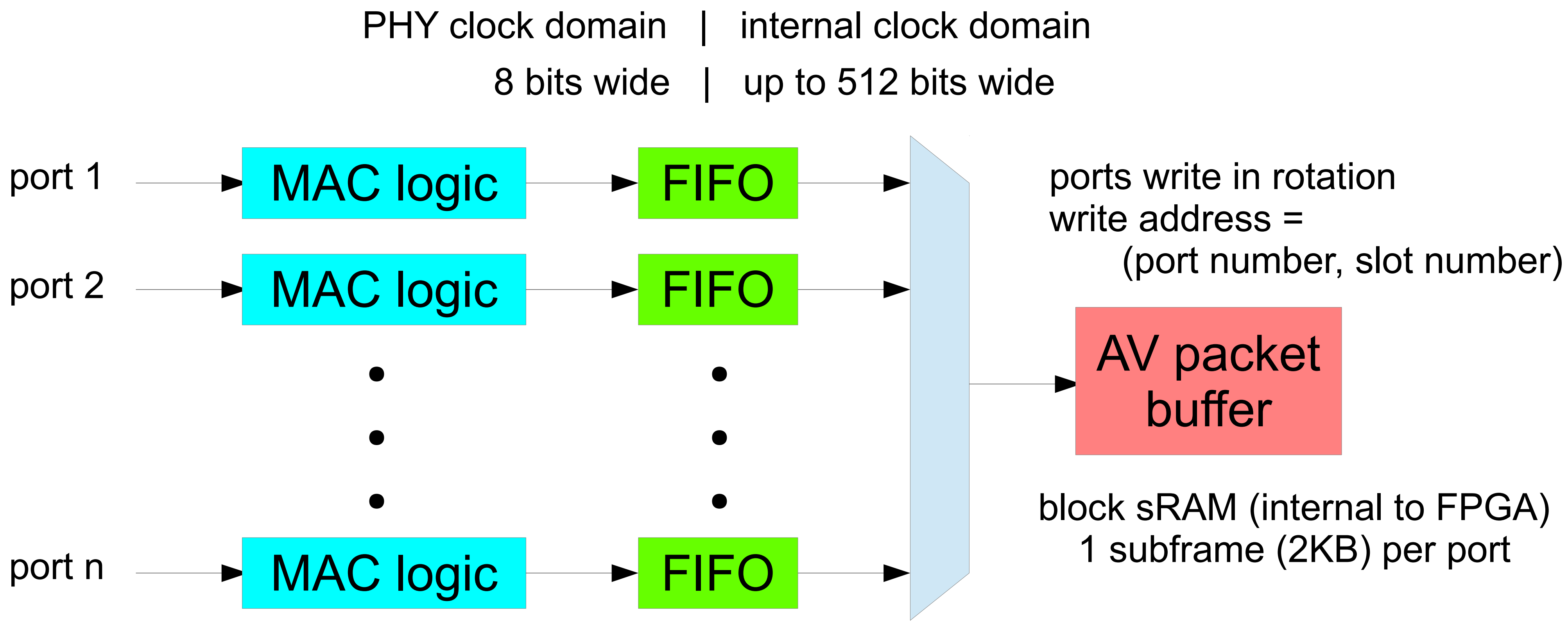
IT packet routing: receive side



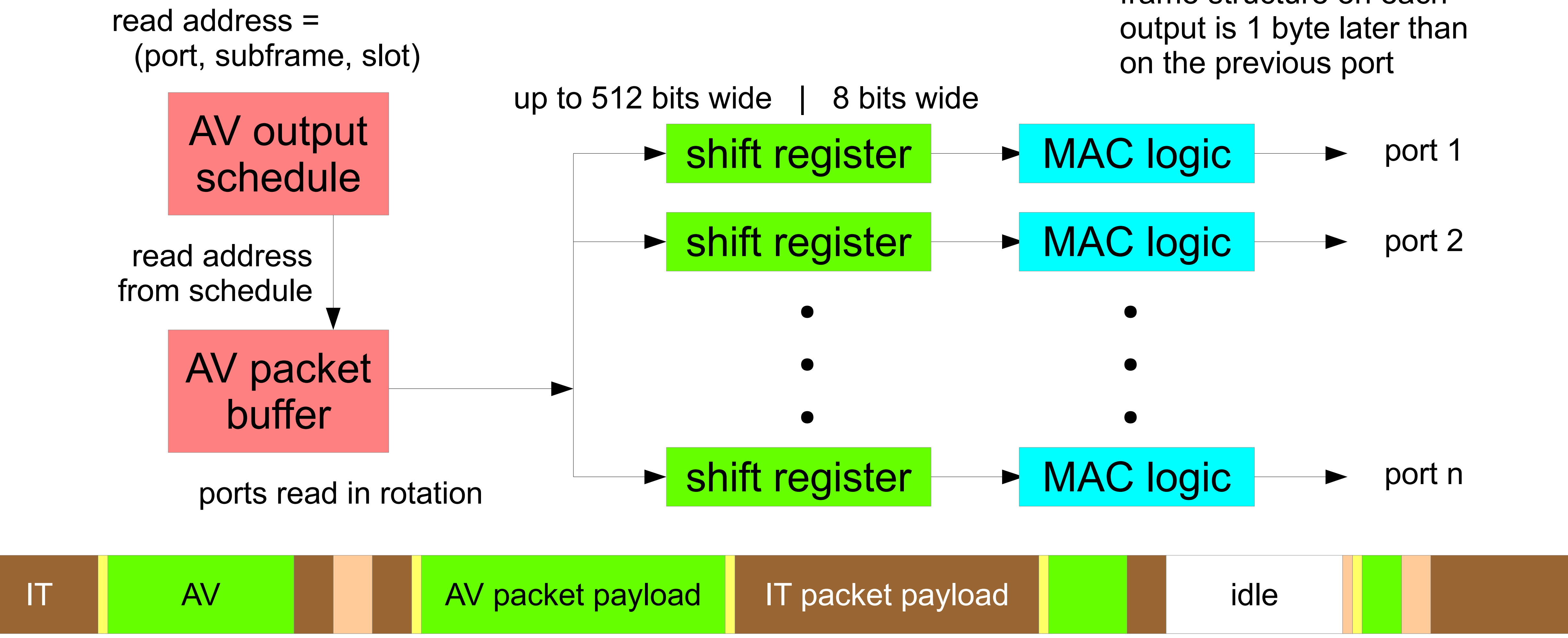
IT packet routing: transmit side



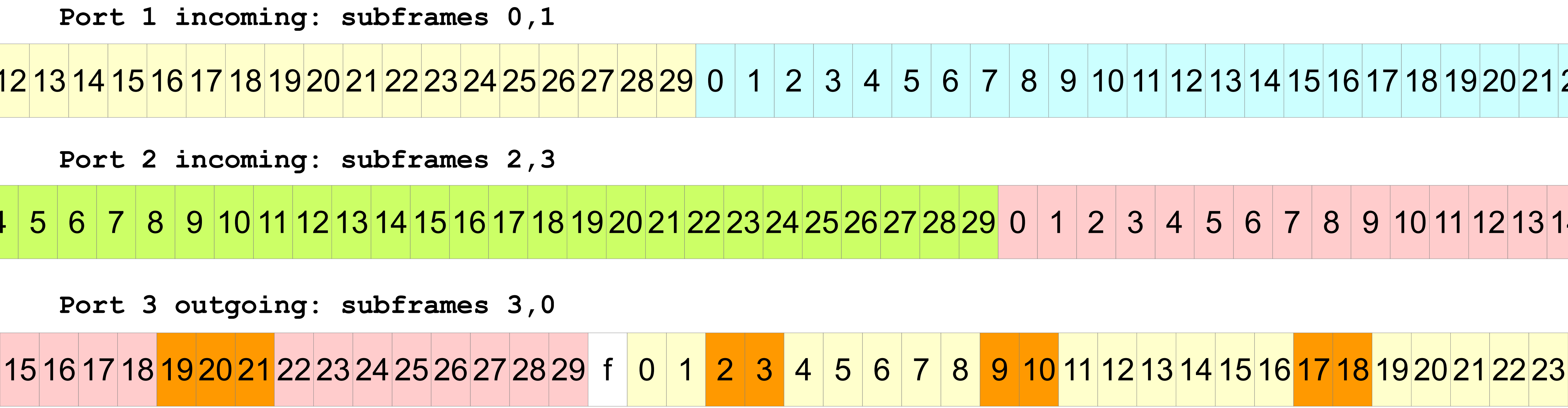
AV packet routing: receive side



AV packet routing: transmit side

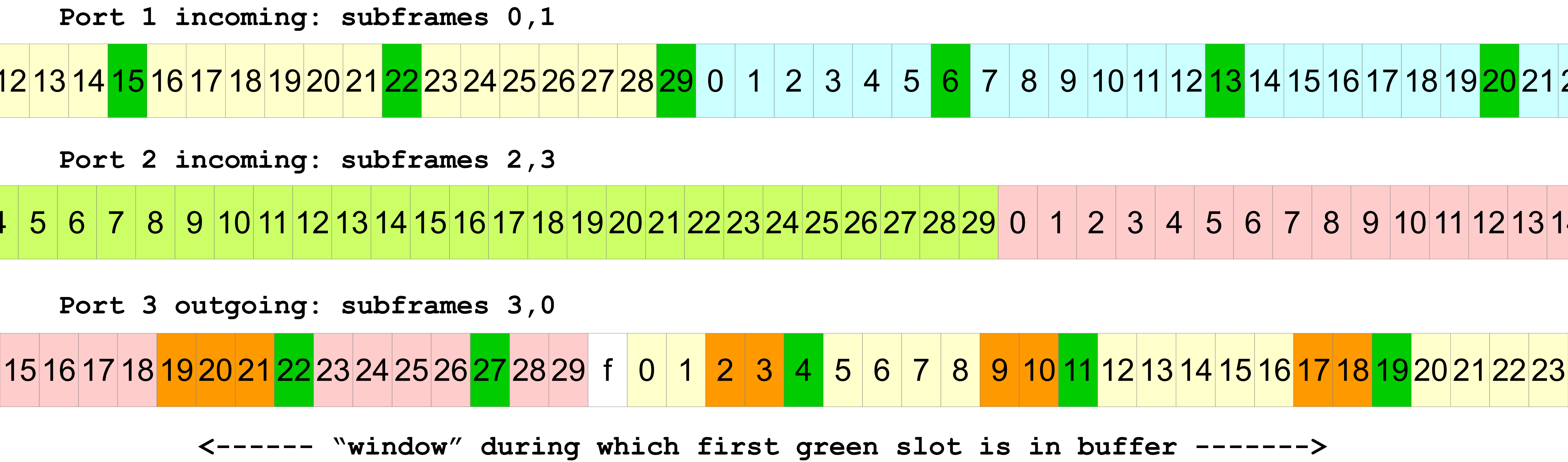


AV packet forwarding example



The orange slots have been allocated to traffic from ports other than 1 and 2

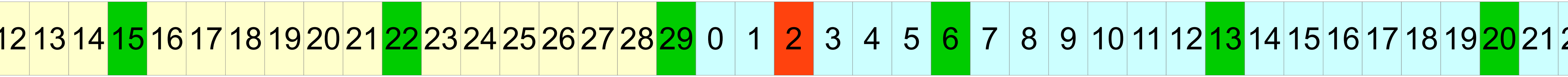
AV packet forwarding example



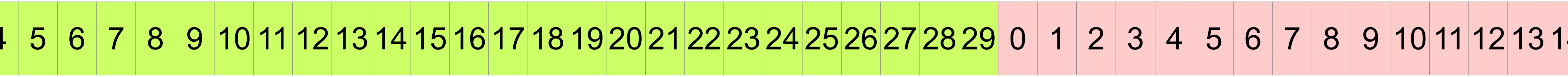
The green flow has been routed, coming in on port 1

AV packet forwarding example

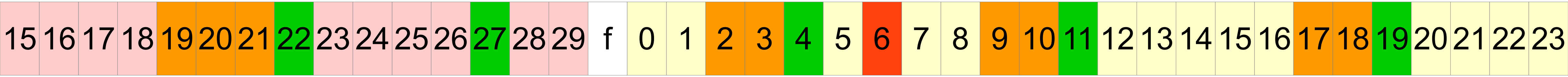
Port 1 incoming: subframes 0,1



Port 2 incoming: subframes 2,3



Port 3 outgoing: subframes 3,0



The red flow has been routed, coming in on port 1

AV packet forwarding example

`Port 1 incoming: subframes 0,1`

12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 2

`Port 2 incoming: subframes 2,3`

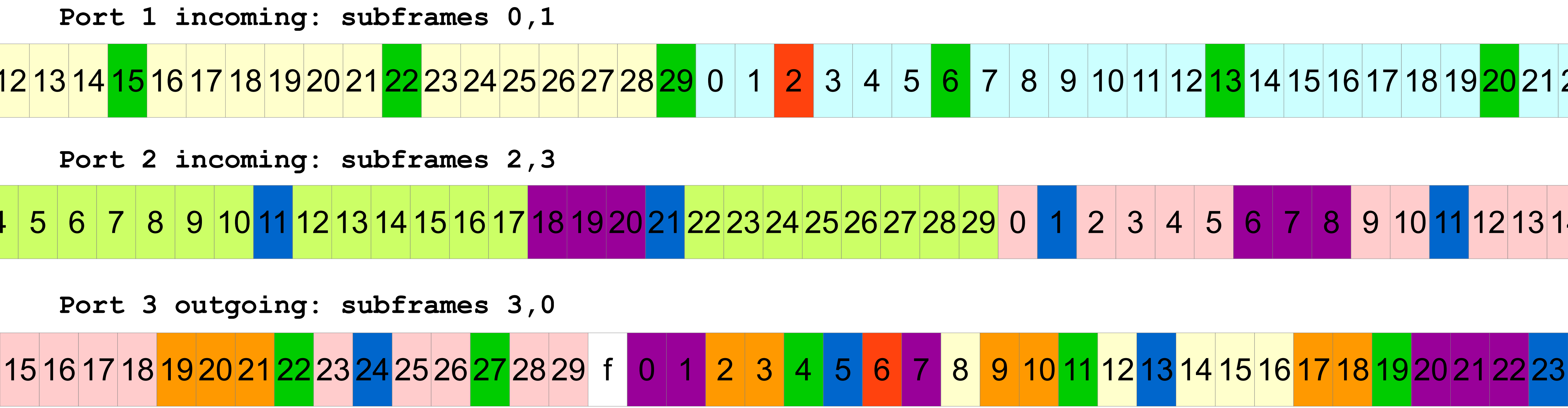
4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 0 1 2 3 4 5 6 7 8 9 10 11 12 13 1

`Port 3 outgoing: subframes 3,0`

15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 f 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

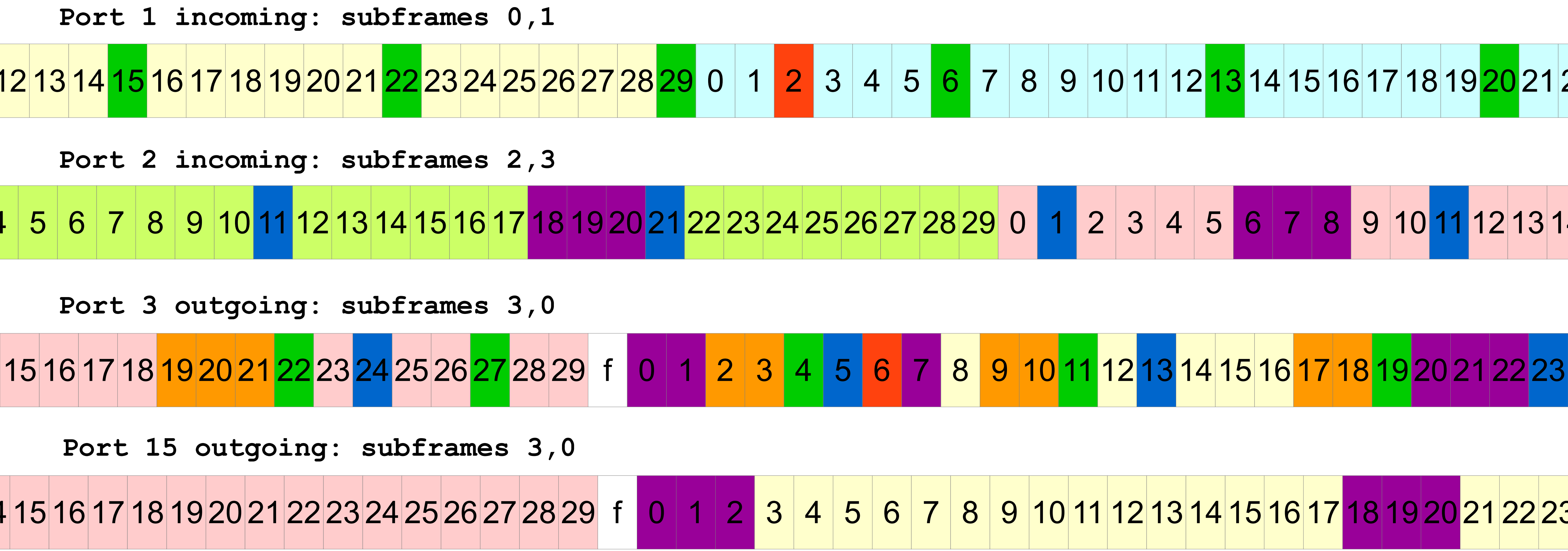
The blue flow has been routed, coming in on port 2

AV packet forwarding example



The purple flow has been routed, coming in on port 2

AV packet forwarding example



The purple flow has also been routed to port 15, which didn’t have any flows routed previously