AMChip06: circuit de mémoire associative en technologie TSMC 65 nm pour l'upgrade FTK d'ATLAS

F. Crescioli for the AMchip Team

LPNHE PARIS

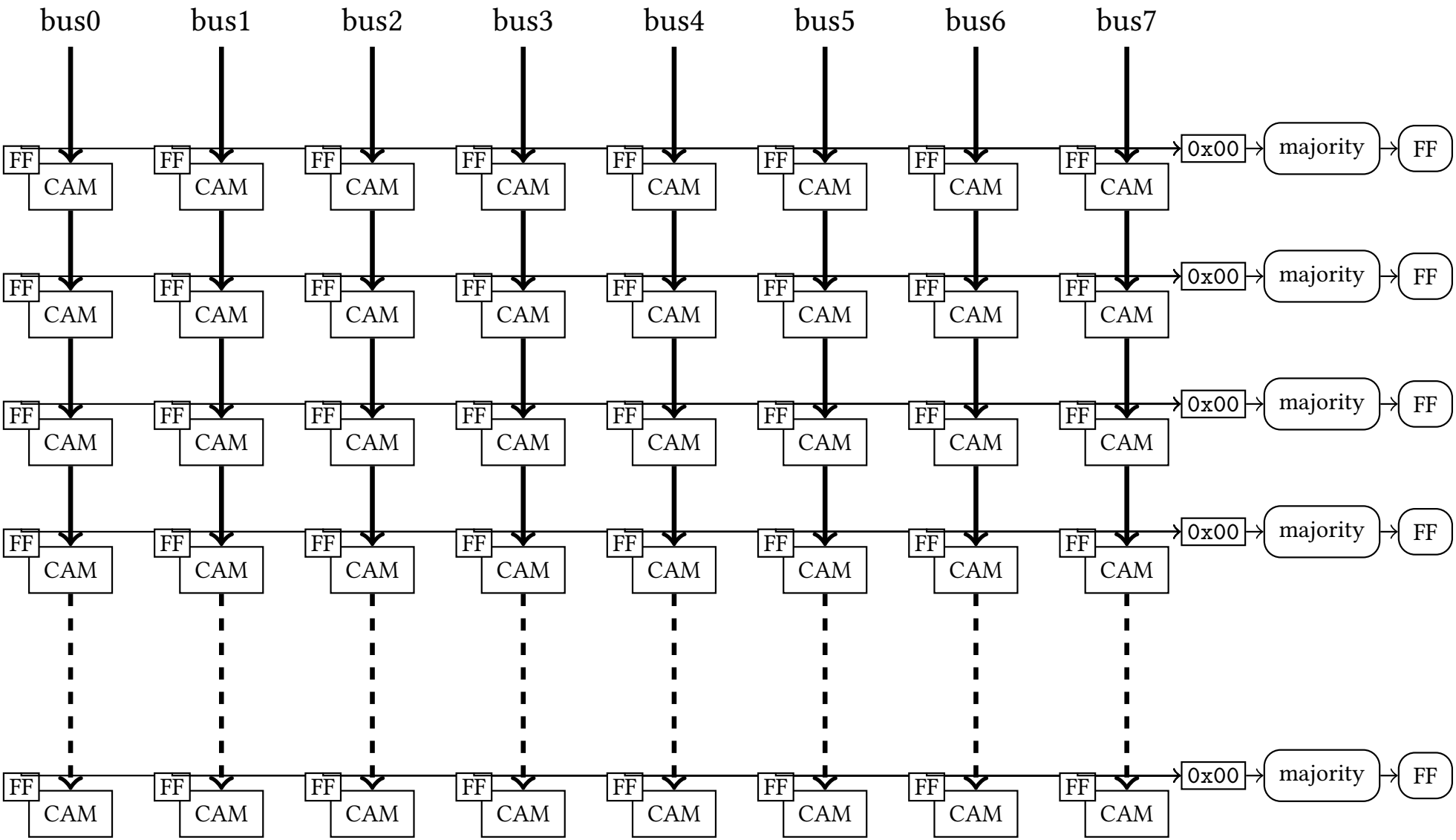
IPHC Strasbourg - 01-06-2016

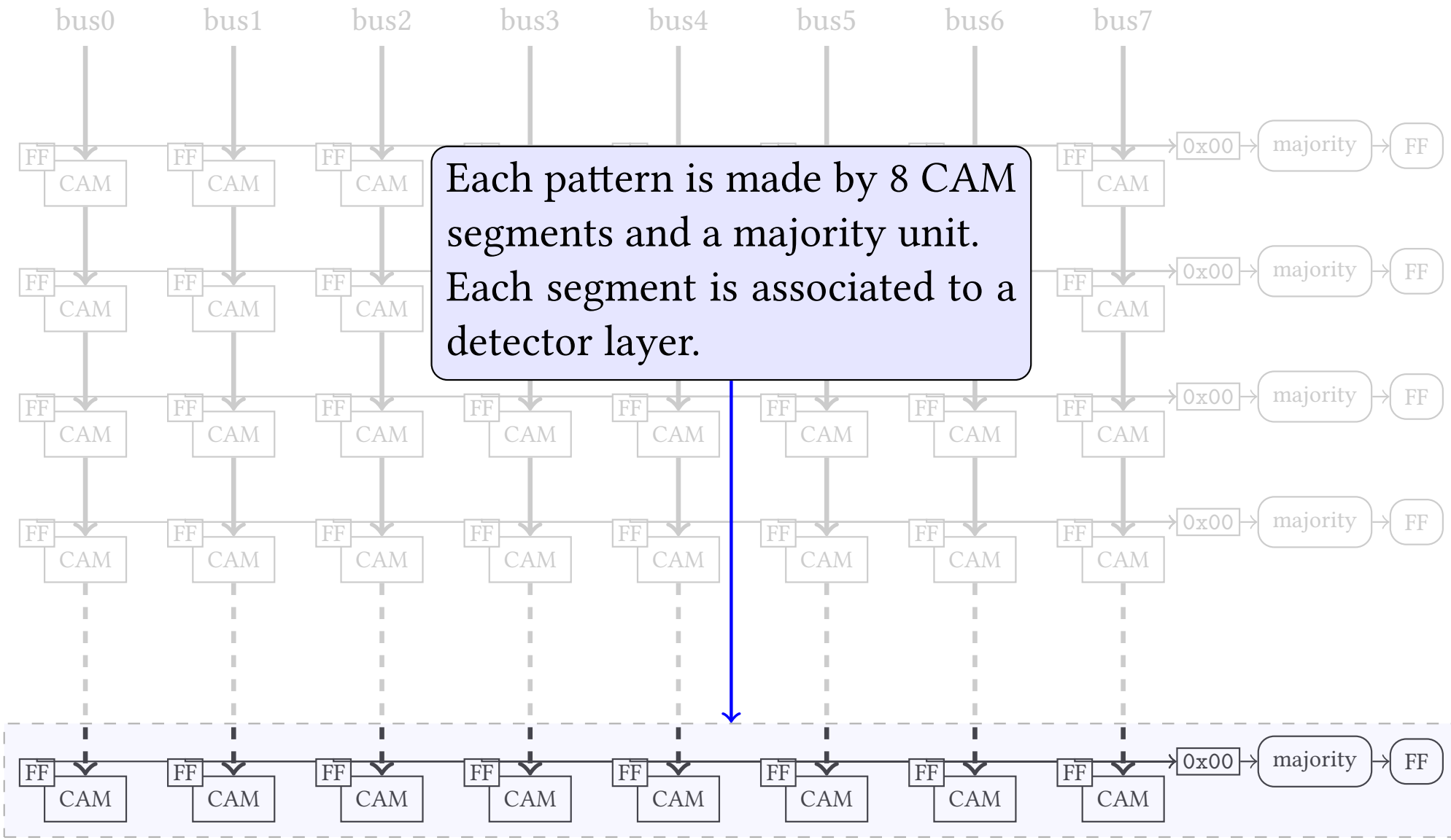
Outline

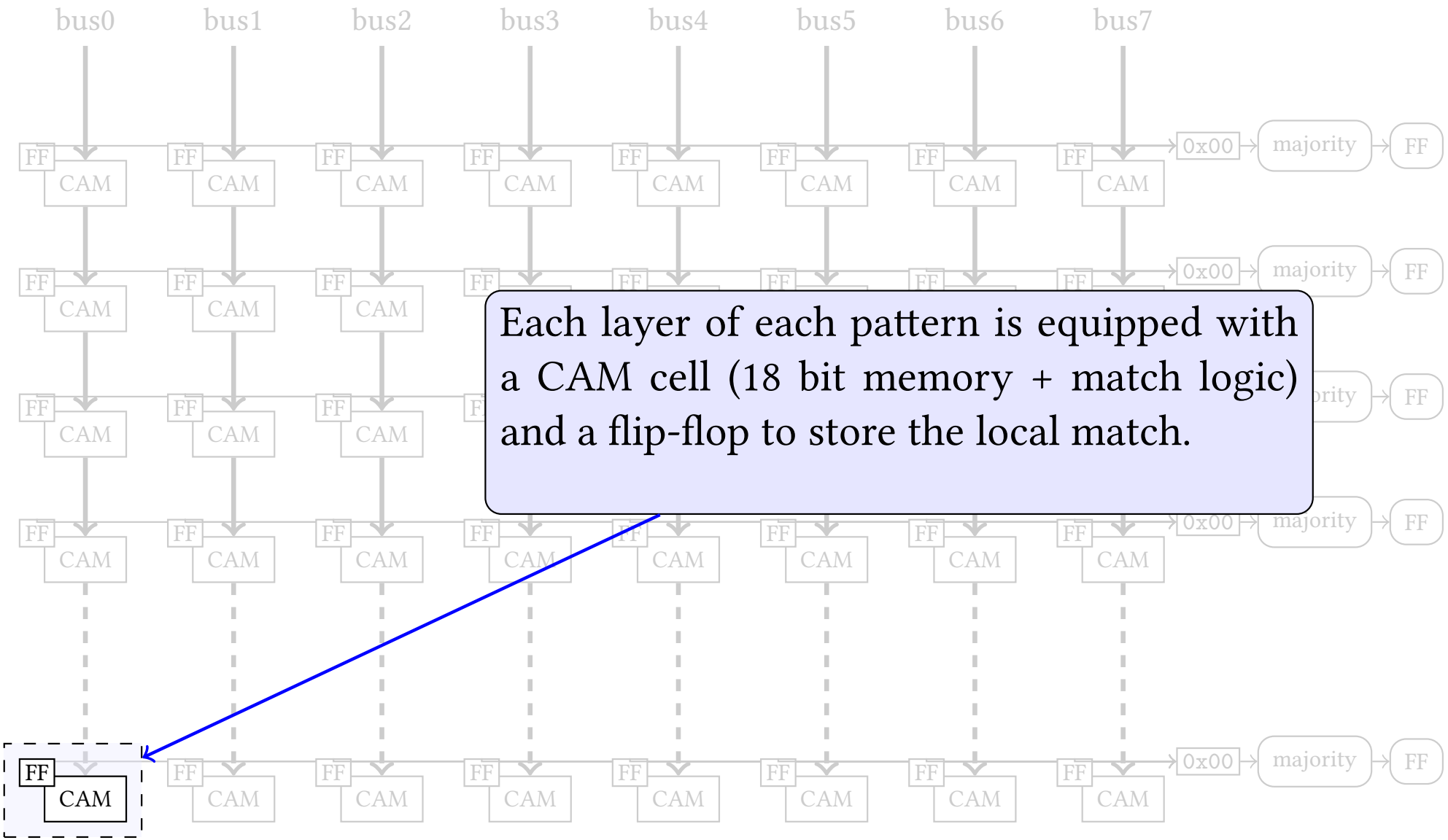
- Introduction to the Associative Memory
- Design approach
- Remarks on the flow
- Conclusions

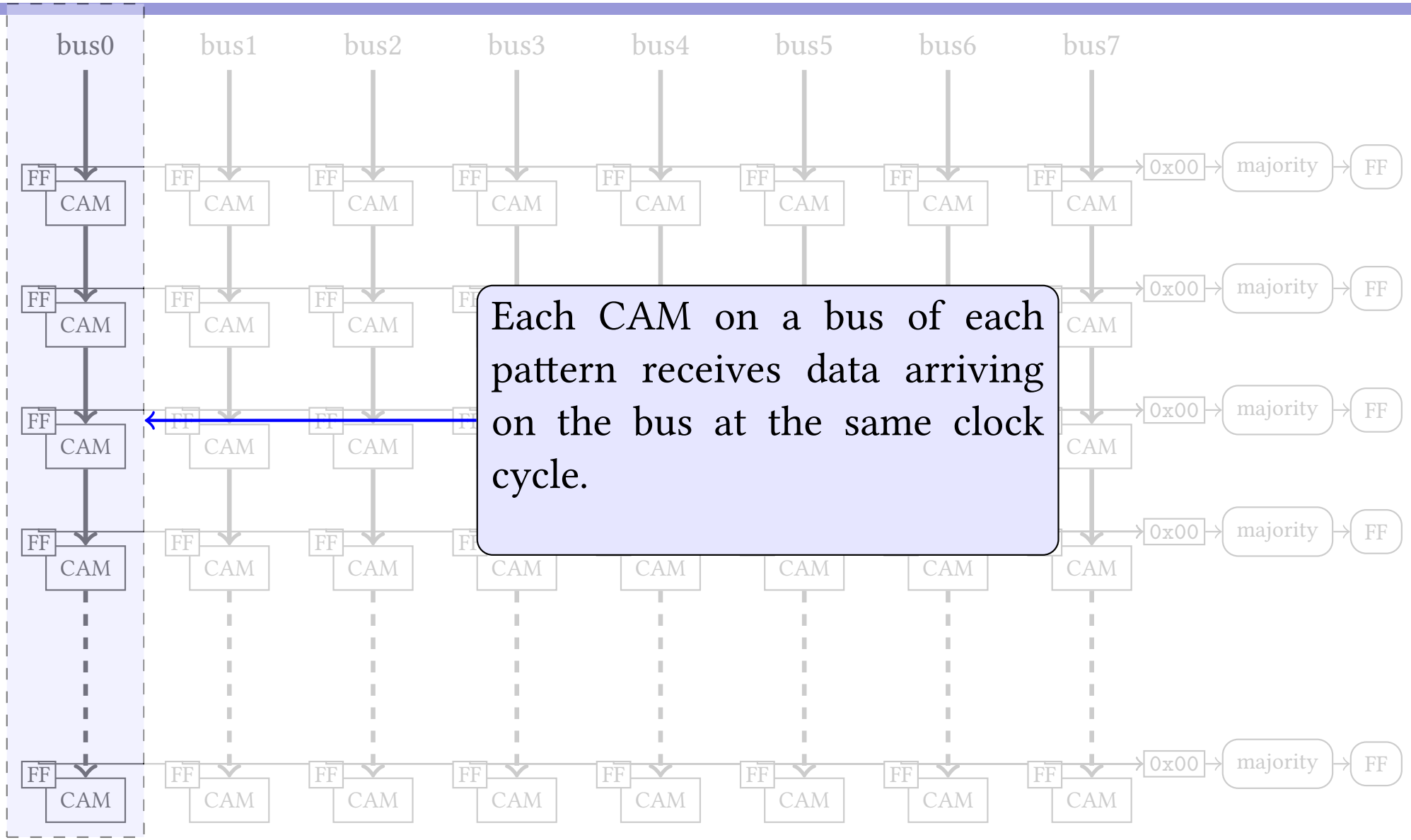
Associative Memory

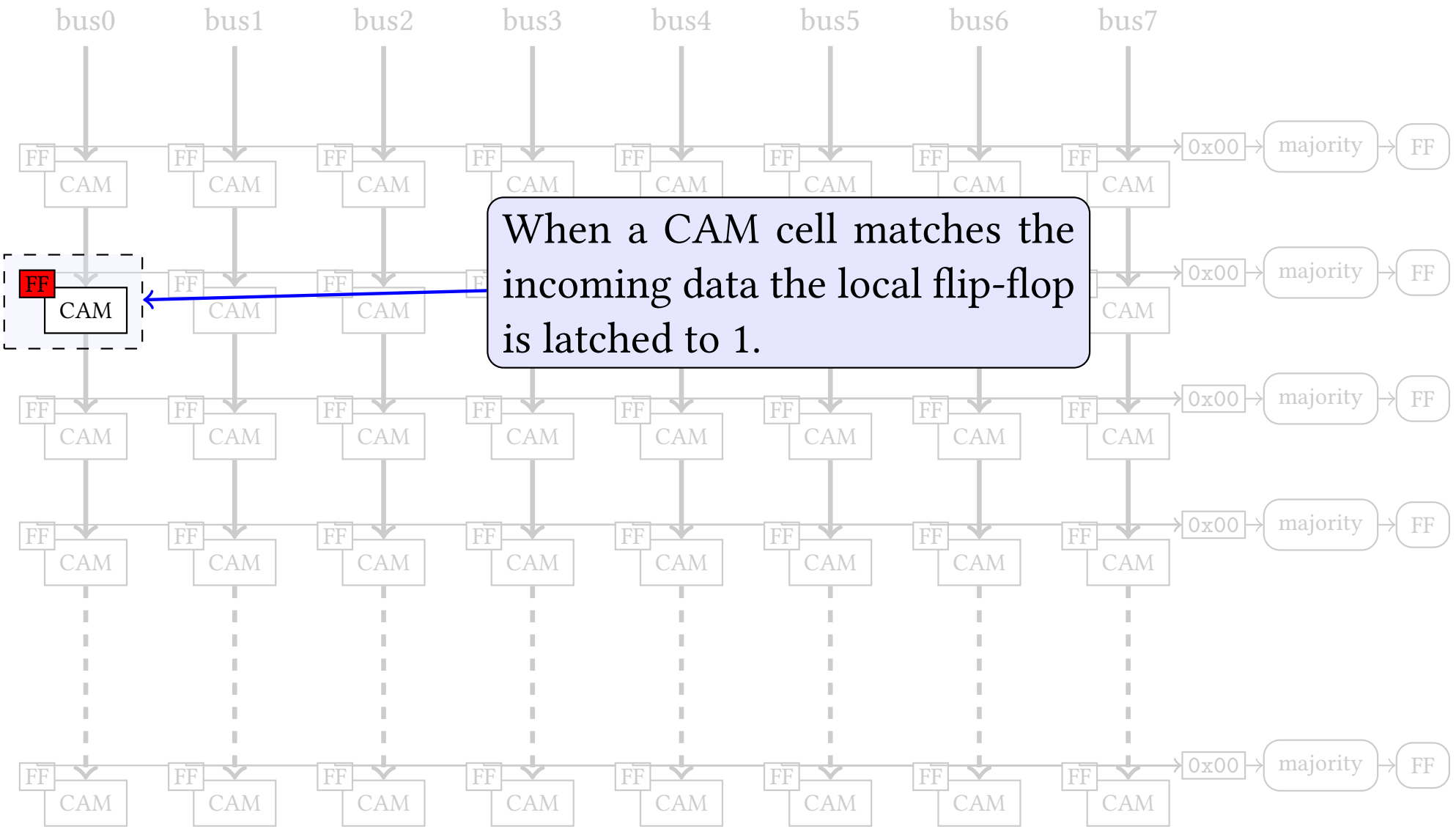
- The Associative Memory is an original computing device first conceived for realtime reconstruction of the trajectories of charged particles (tracking) at the CDF experiment of the hadron collider Tevatron
- The Associative Memory finds all matches between all combinations of input data and a pre-loaded database of patterns. It is a combinatorial pattern recognition engine.
- The AMchip, the ASIC that implements the Associative Memory function, is currently at the AMchip06 version (TSMC 65 nm) developed for ATLAS FTK
- AMchip06 has been developed by INFN (Milano, Frascati, Pisa) and LPNHE (Paris)

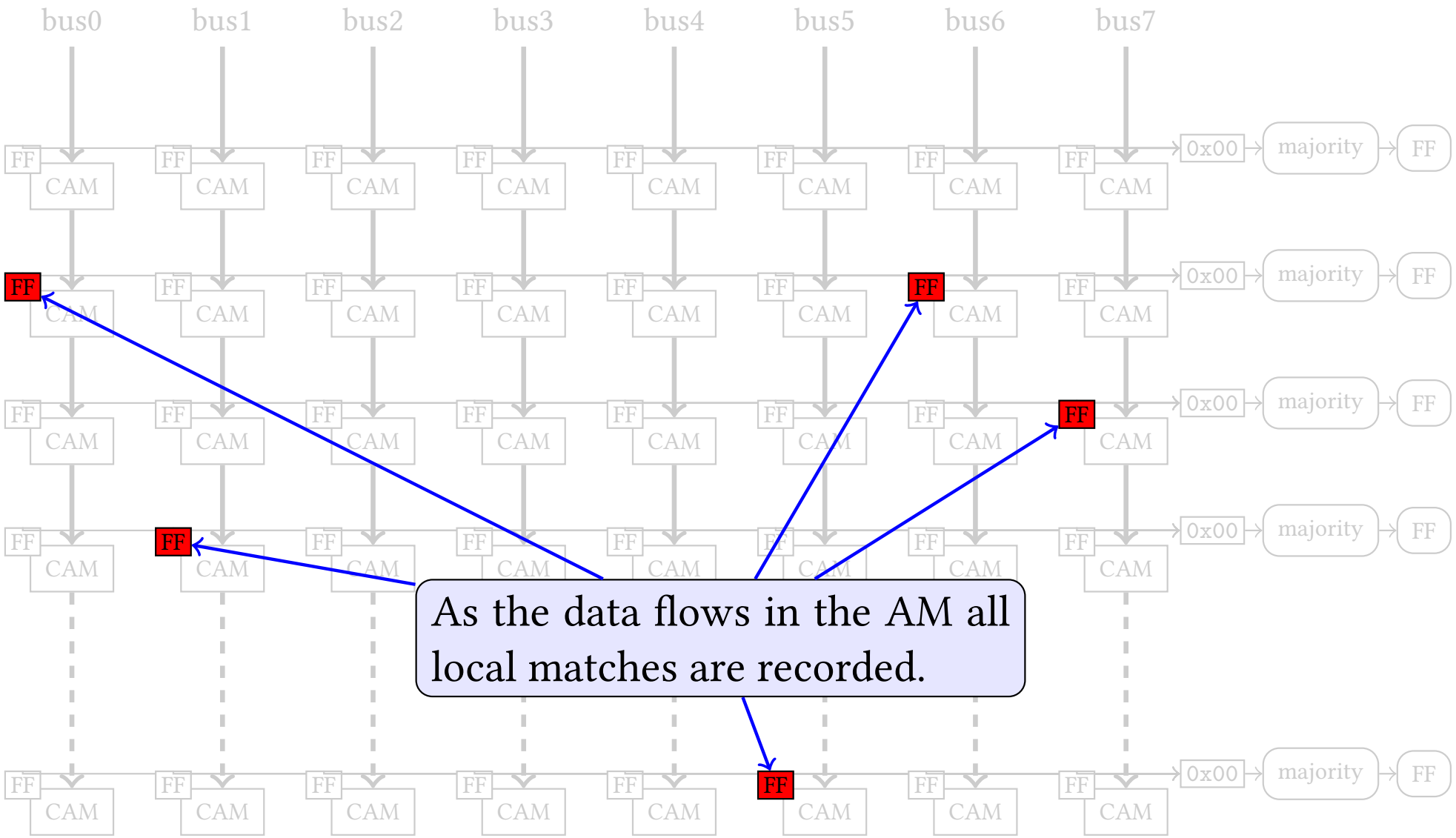

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF

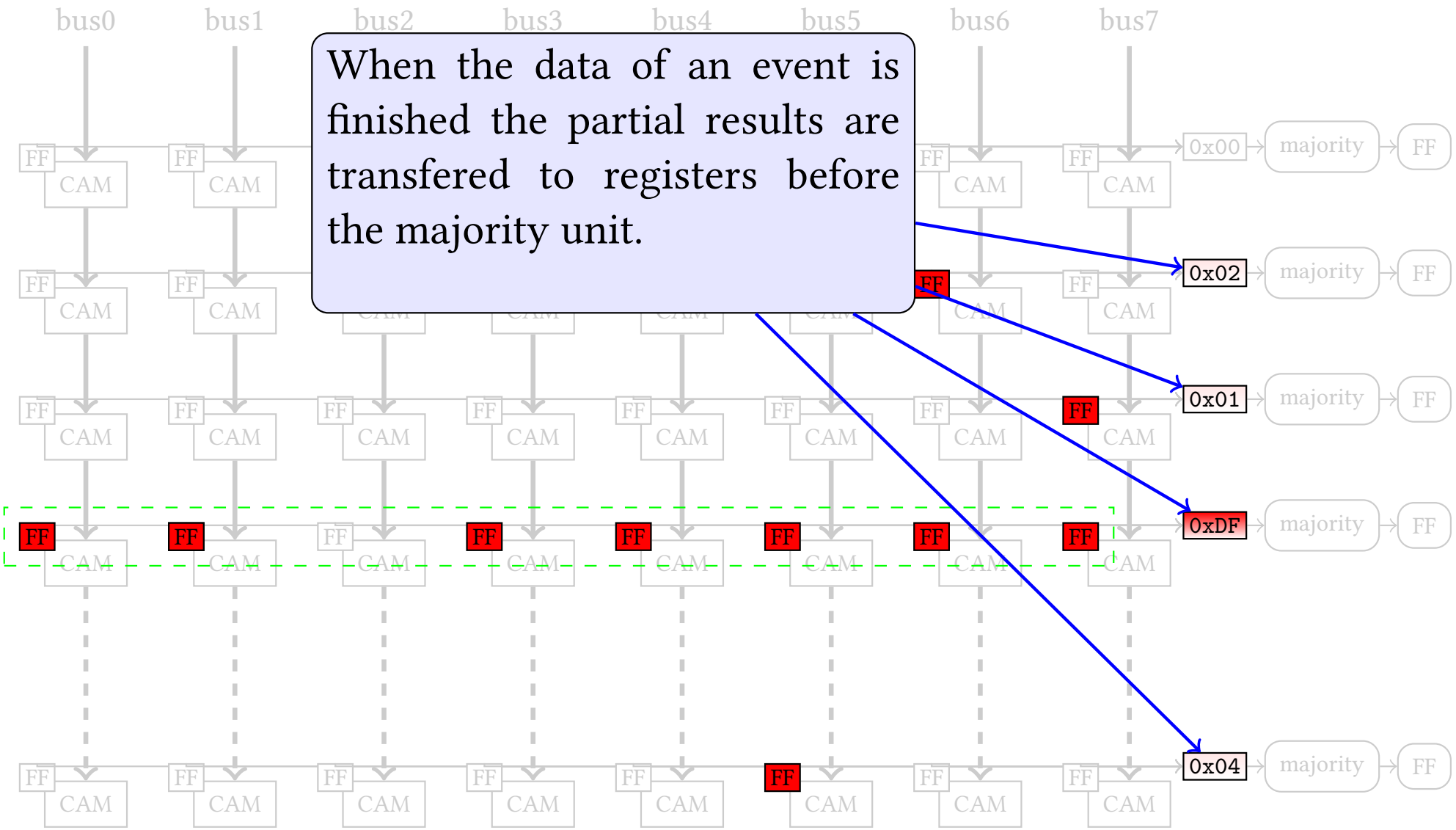

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF
Each pattern is made by 8 CAM segments and a majority unit. Each segment is associated to a detector layer.

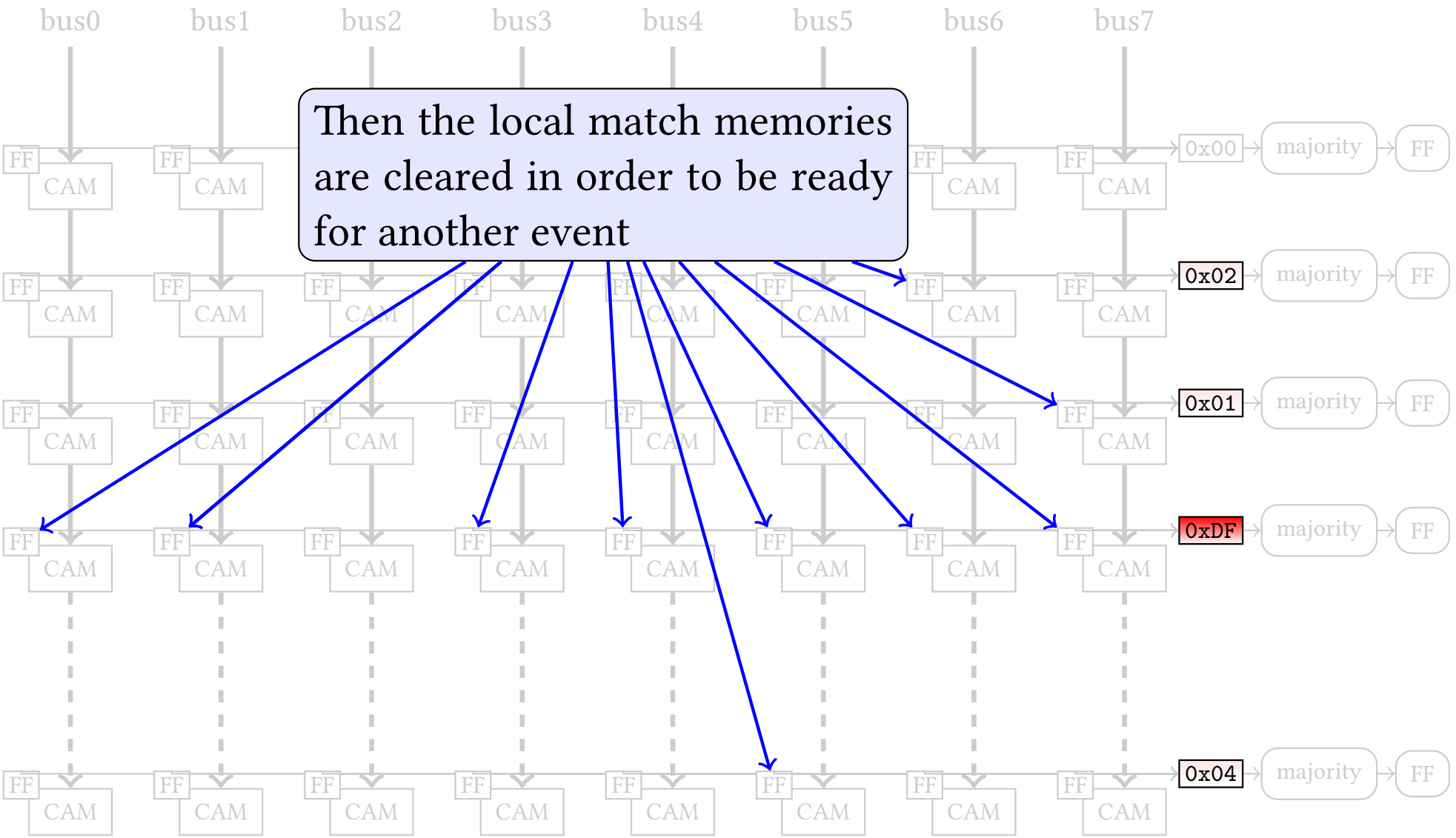

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF
Each layer of each pattern is equipped with a CAM cell (18 bit memory + match logic) and a flip-flop to store the local match.

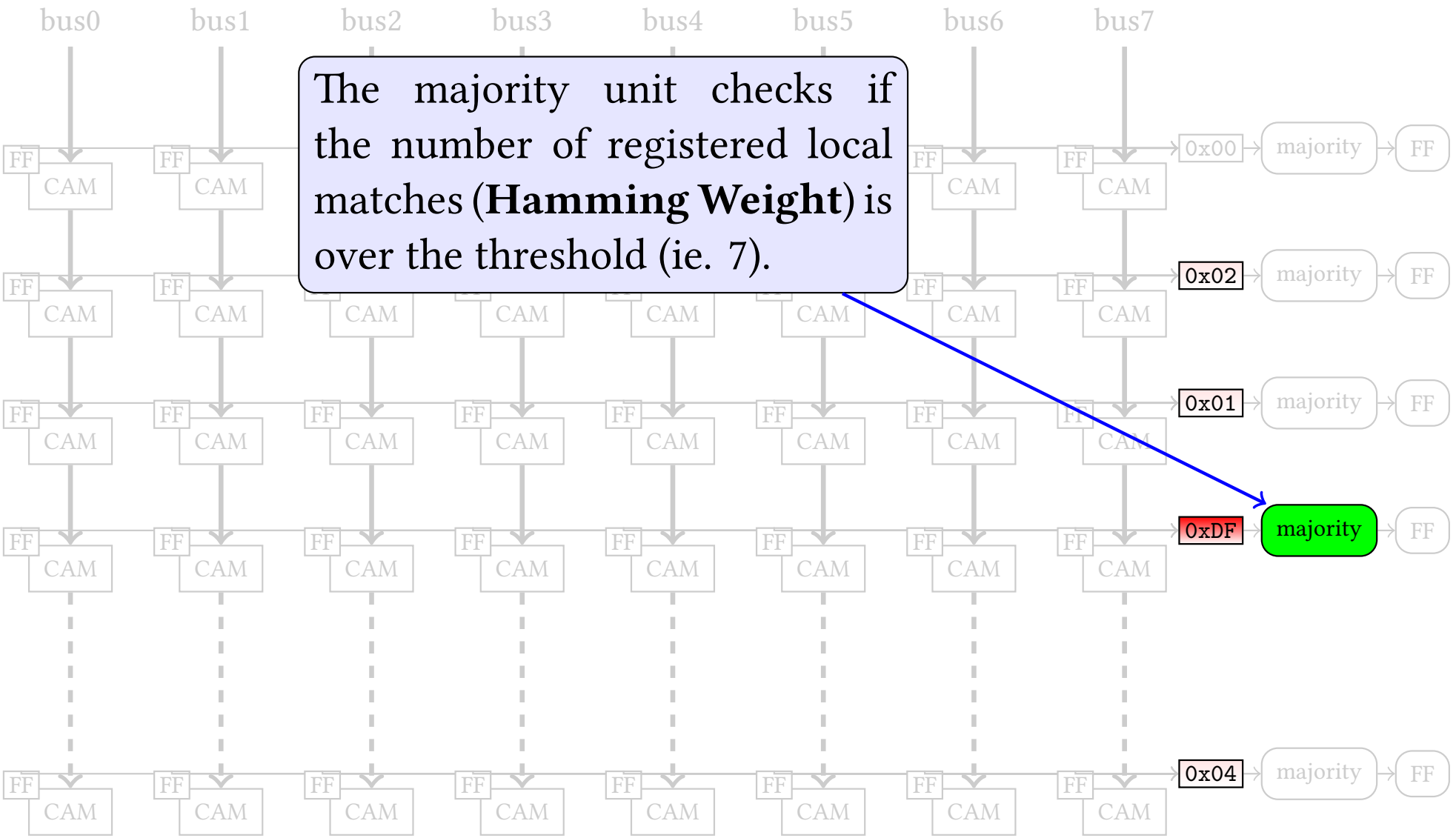

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF
Each CAM on a bus of each pattern receives data arriving on the bus at the same clock cycle.


bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
When a CAM cell matches the incoming data the local flip-flop is latched to 1.

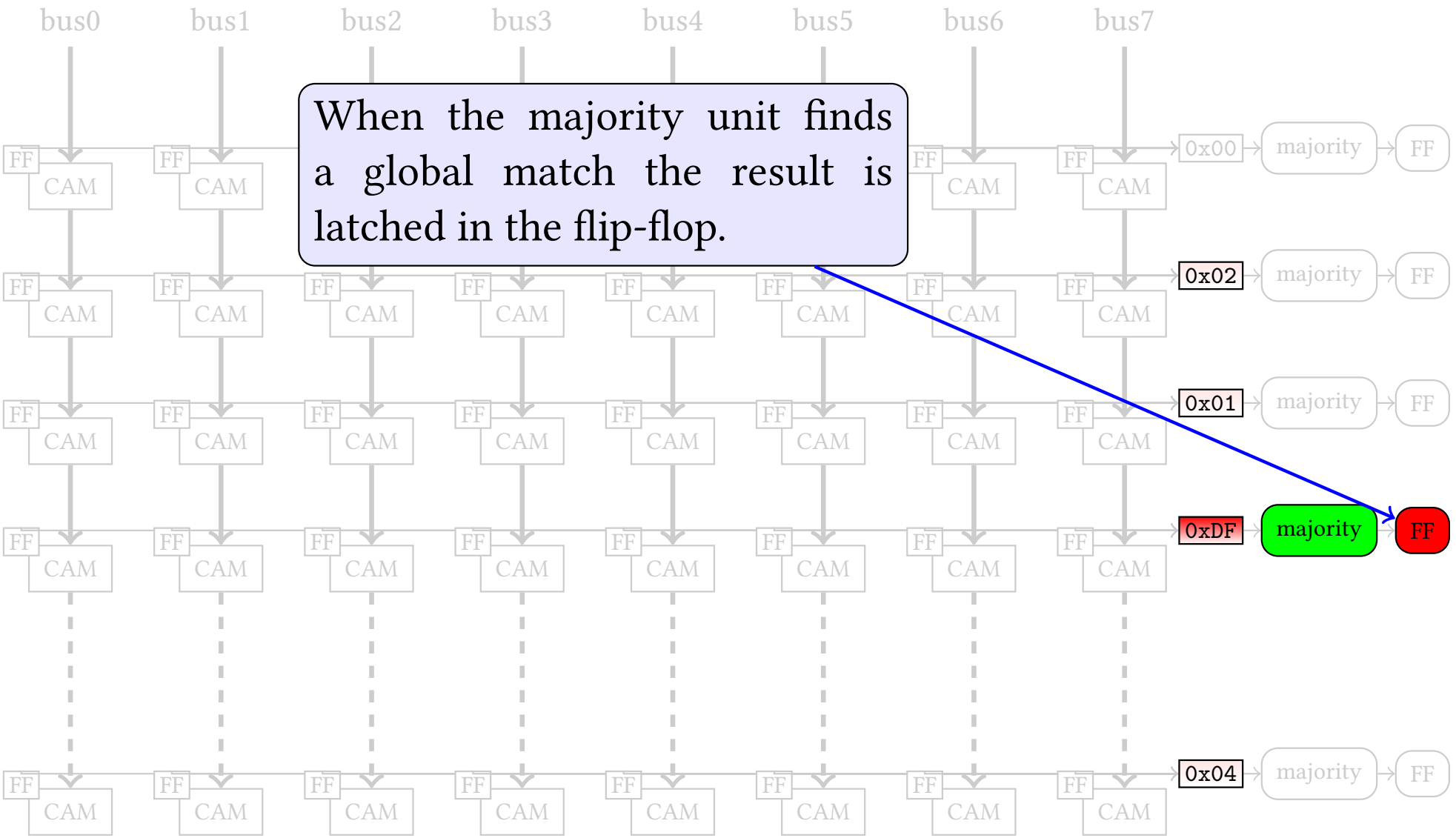

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
As the data flows in the AM all local matches are recorded.

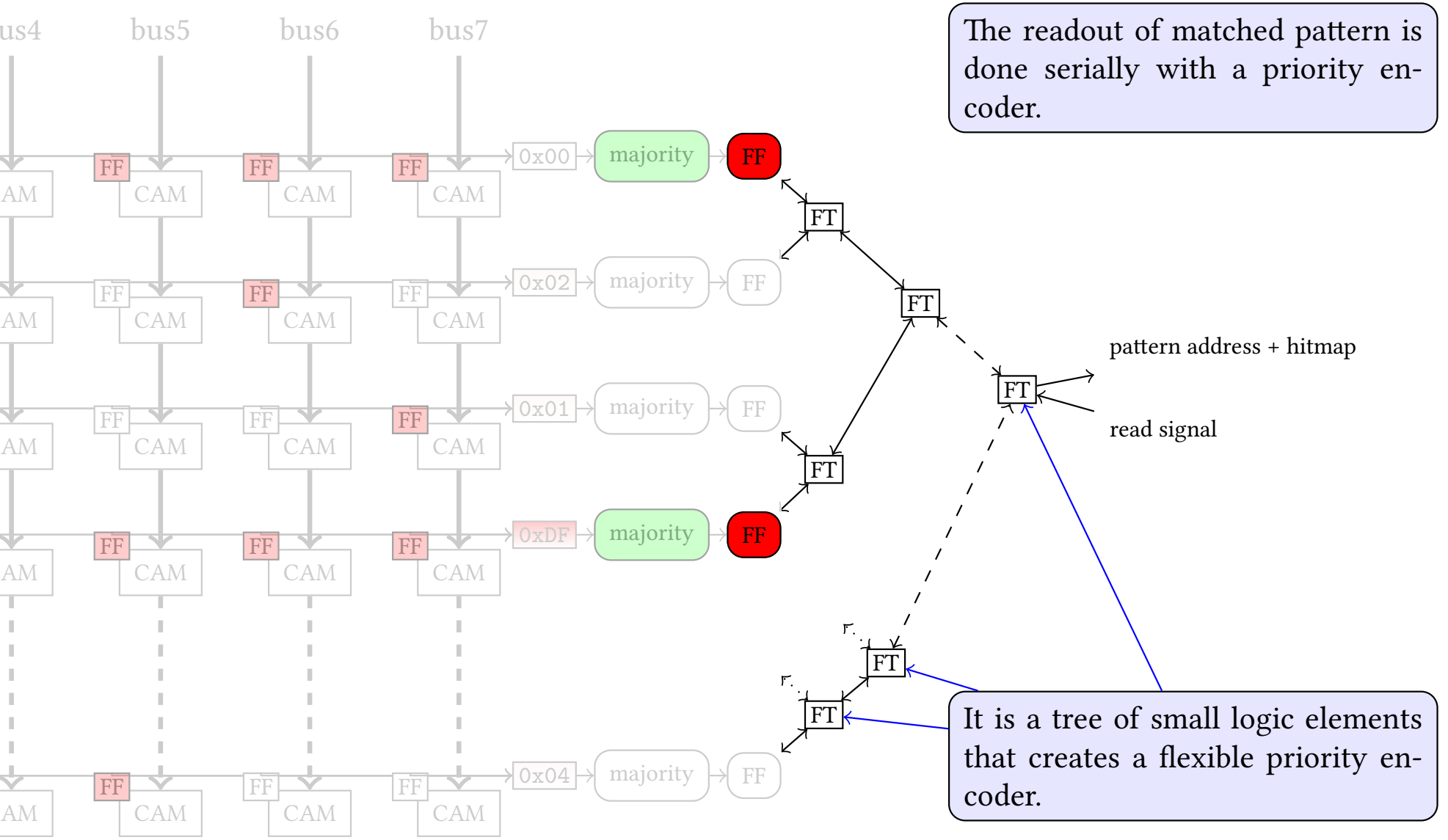

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
When the data of an event is finished the partial results are transfered to registers before the majority unit.
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority


bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
Then the local match memories are cleared in order to be ready for another event
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority
FF



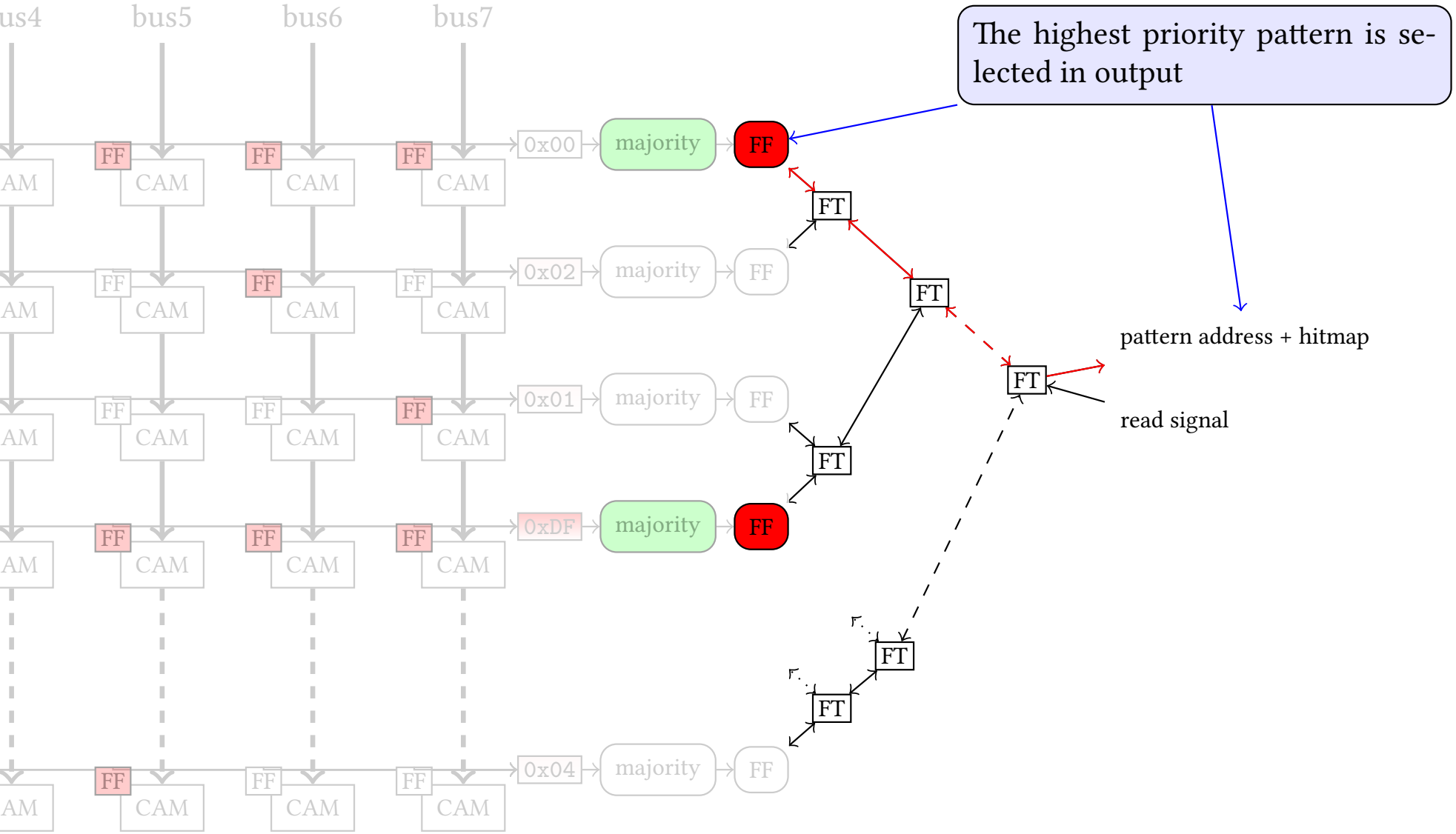
The majority unit checks if the number of registered local matches (**Hamming Weight**) is over the threshold (ie. 7).

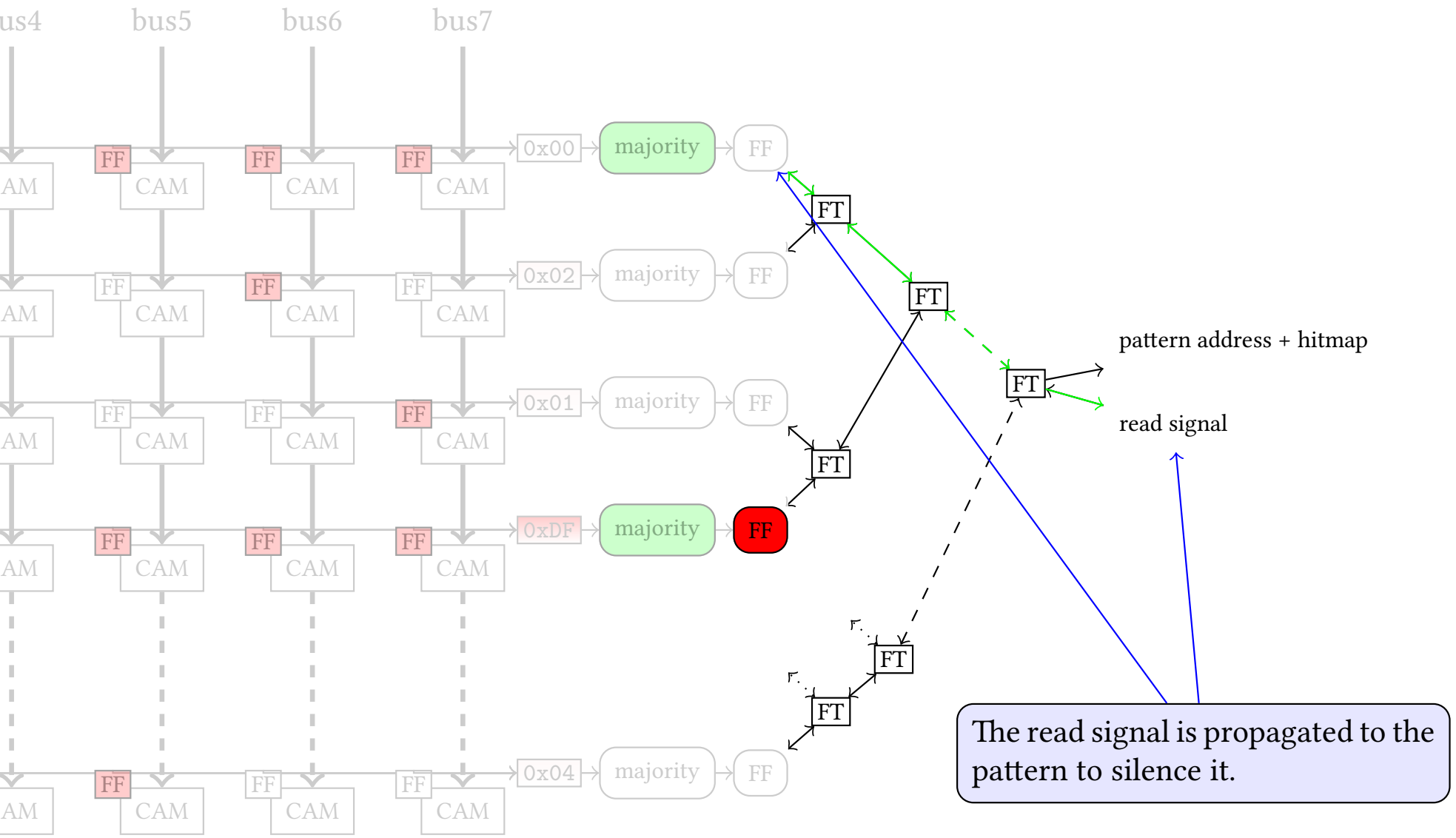

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
When the majority unit finds a global match the result is latched in the flip-flop.
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority
FF

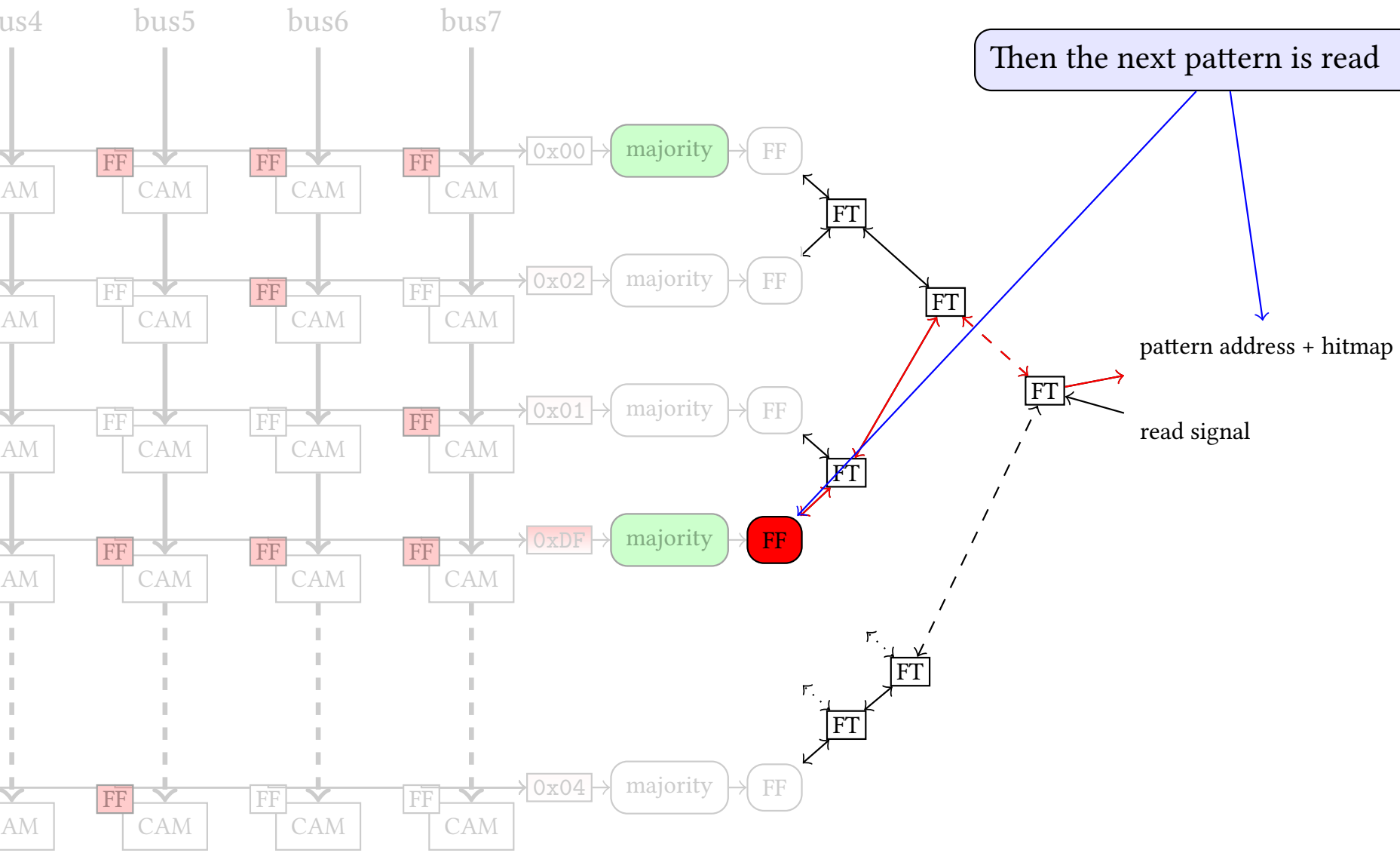


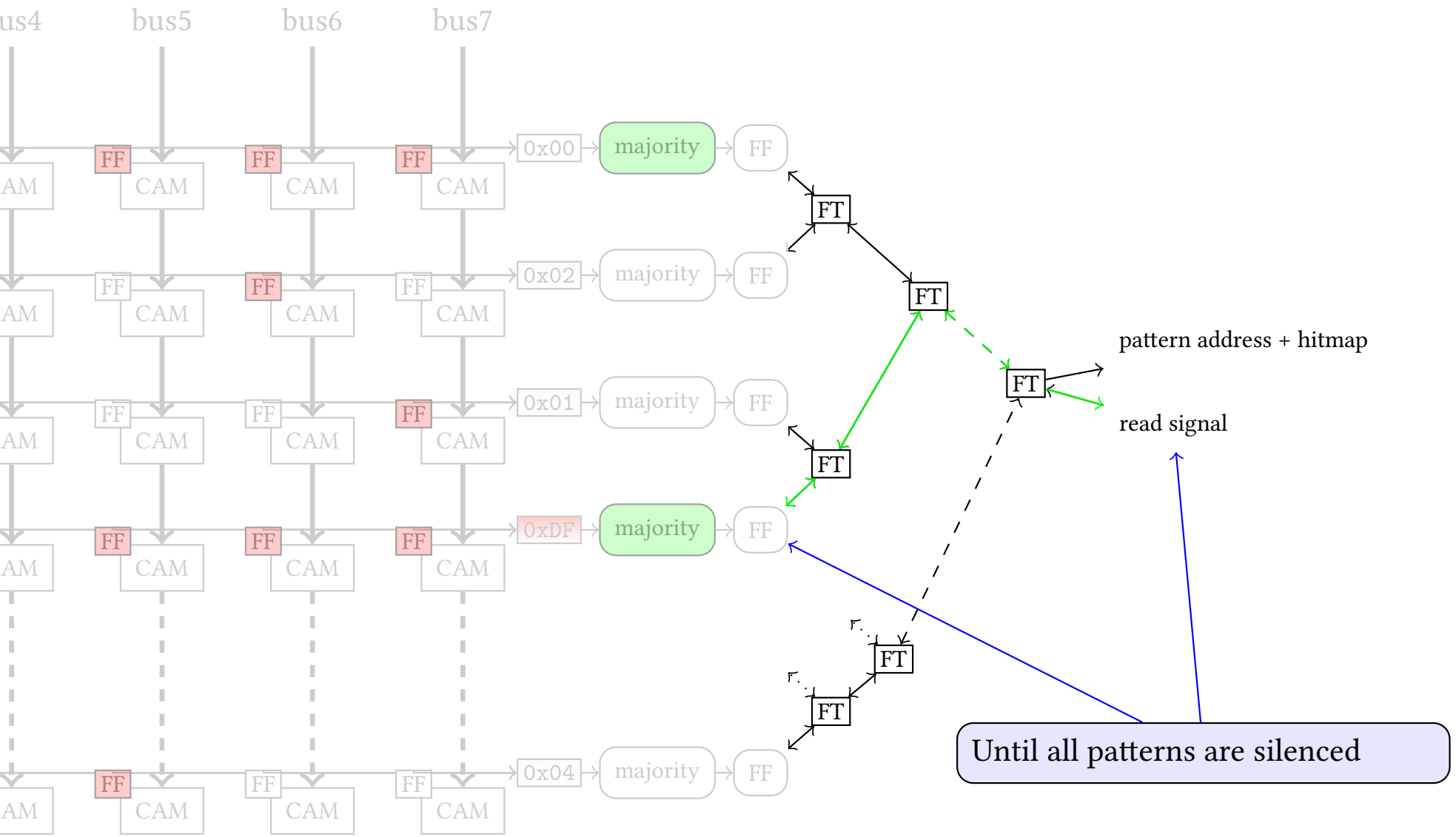
The readout of matched pattern is done serially with a priority encoder.

It is a tree of small logic elements that creates a flexible priority encoder.

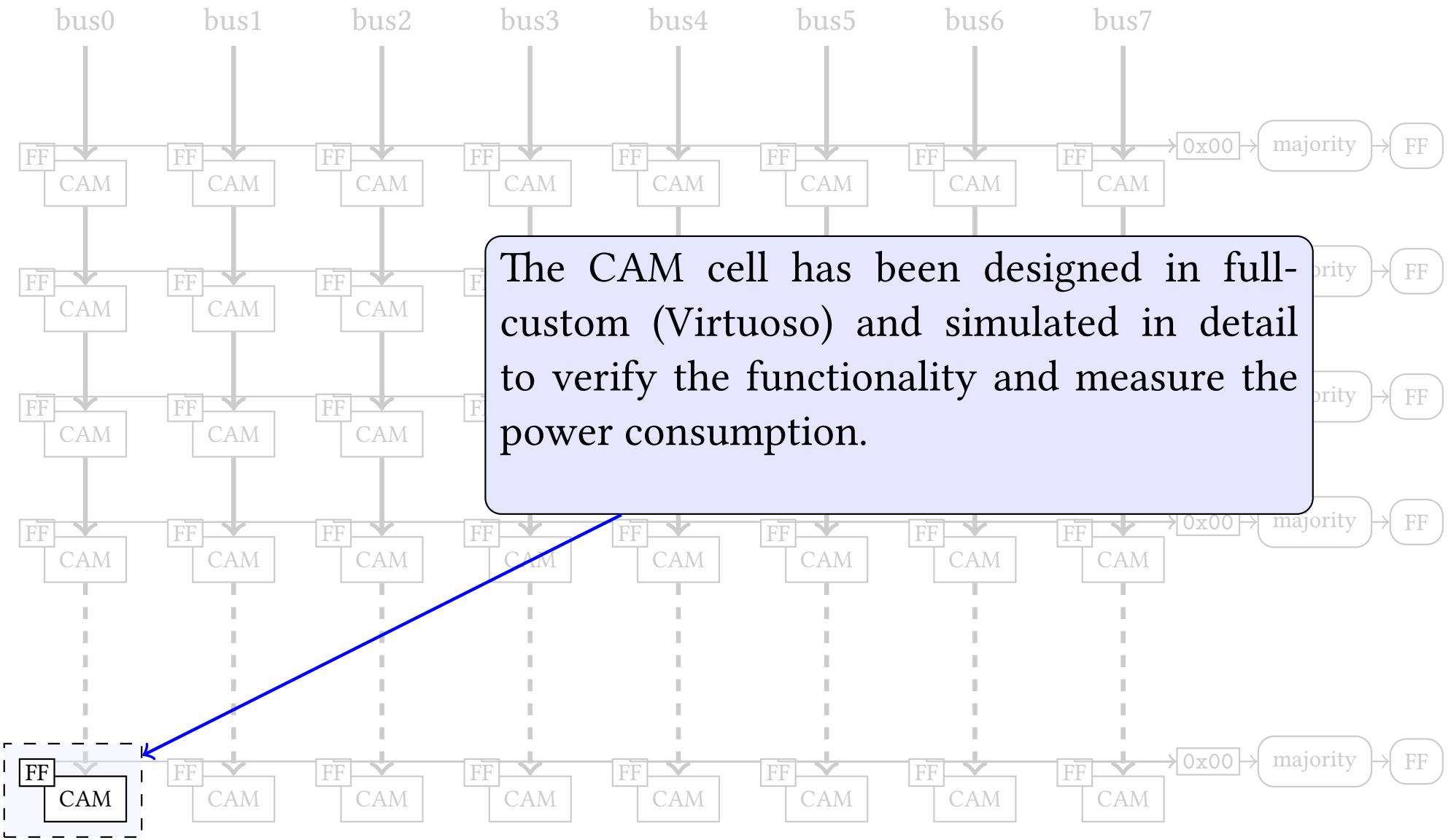

bus4
bus5
bus6
bus7
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority
FT
The highest priority pattern is selected in output
pattern address + hitmap
read signal


bus4
bus5
bus6
bus7
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority
FF
FT
pattern address + hitmap
read signal
The read signal is propagated to the pattern to silence it.


bus4
bus5
bus6
bus7
Then the next pattern is read
FF
CAM
0x00
0x02
0x01
0xDF
0x04
majority
FF
FT
pattern address + hitmap
read signal


us4
bus5
bus6
bus7
FF
CAM
AM
0x00
0x02
0x01
0xDF
0x04
majority
FF
FT
pattern address + hitmap
read signal
Until all patterns are silenced

Design Approach


bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF
The CAM cell has been designed in full-custom (Virtuoso) and simulated in detail to verify the functionality and measure the power consumption.

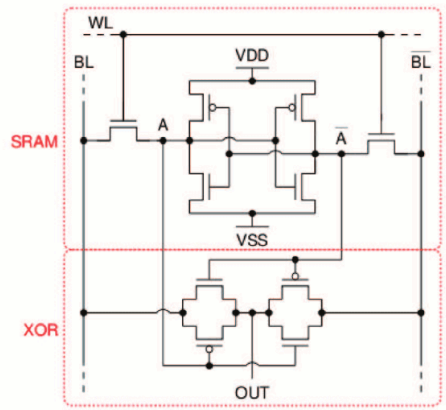


Fig. 6. Schematic diagram of the XOR-based single-bit CAM cell.

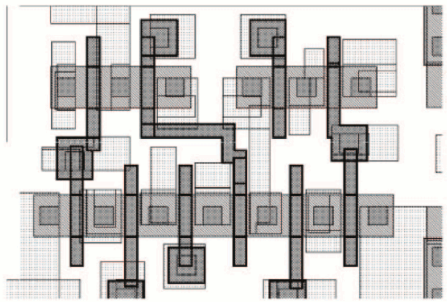


Fig. 7. Layout of the XOR-based CAM cell in 65 nm CMOS technology.

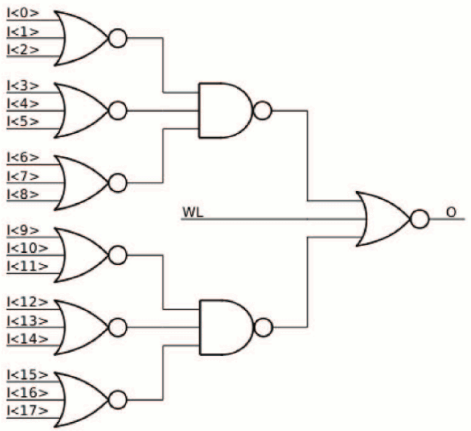
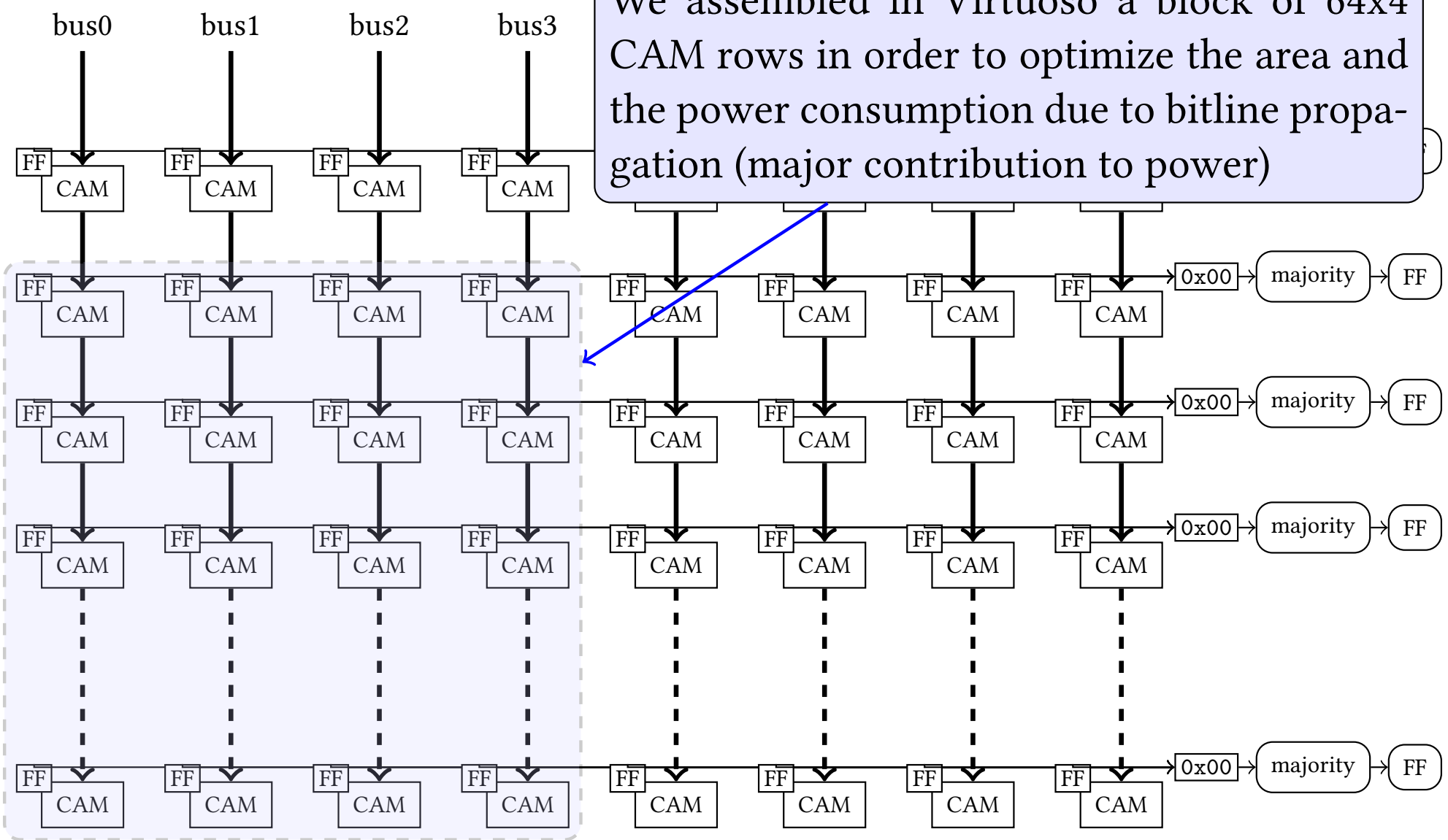
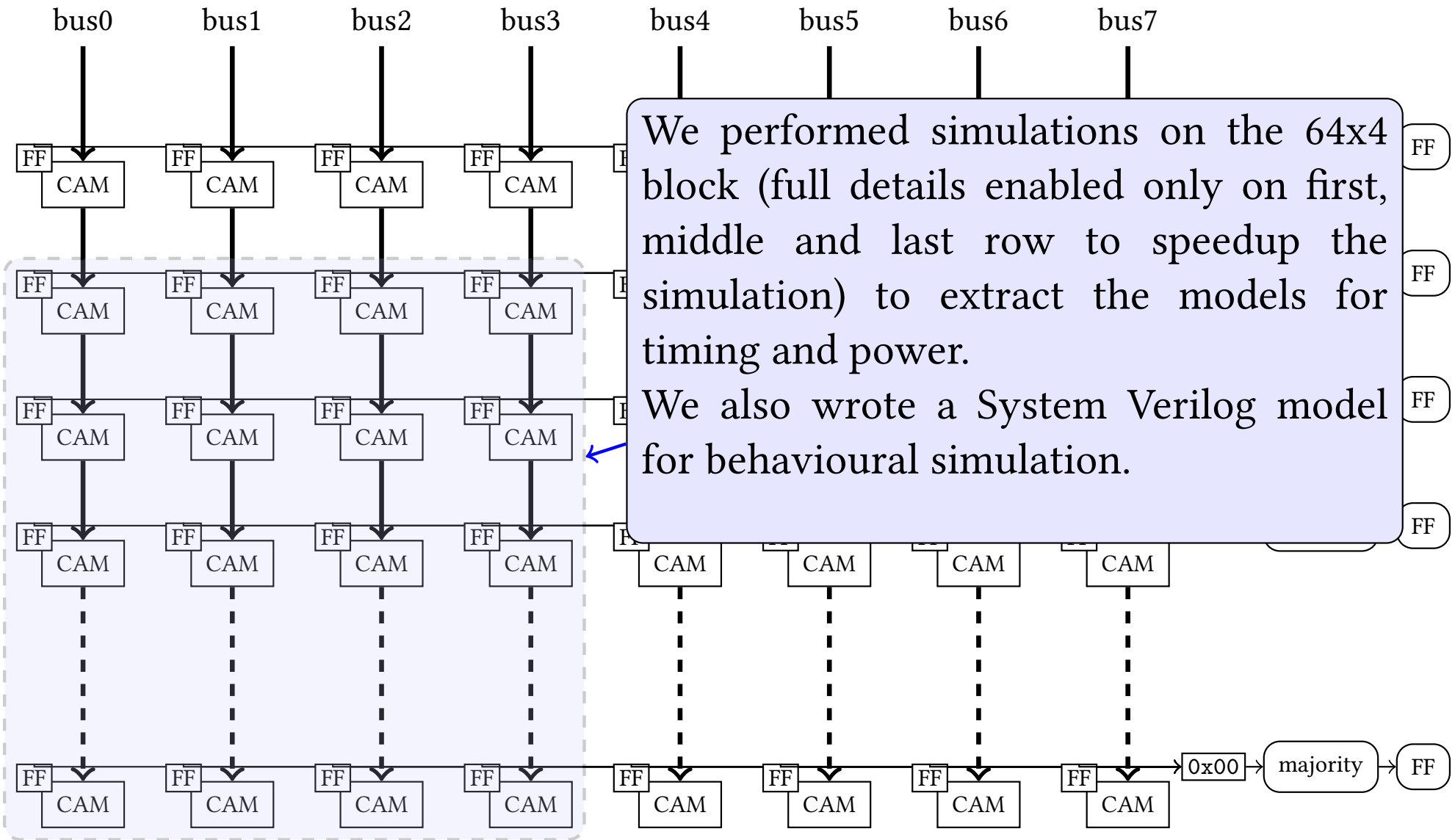


Fig. 8. Schematic diagram of the 18-bit NOR logic gate.

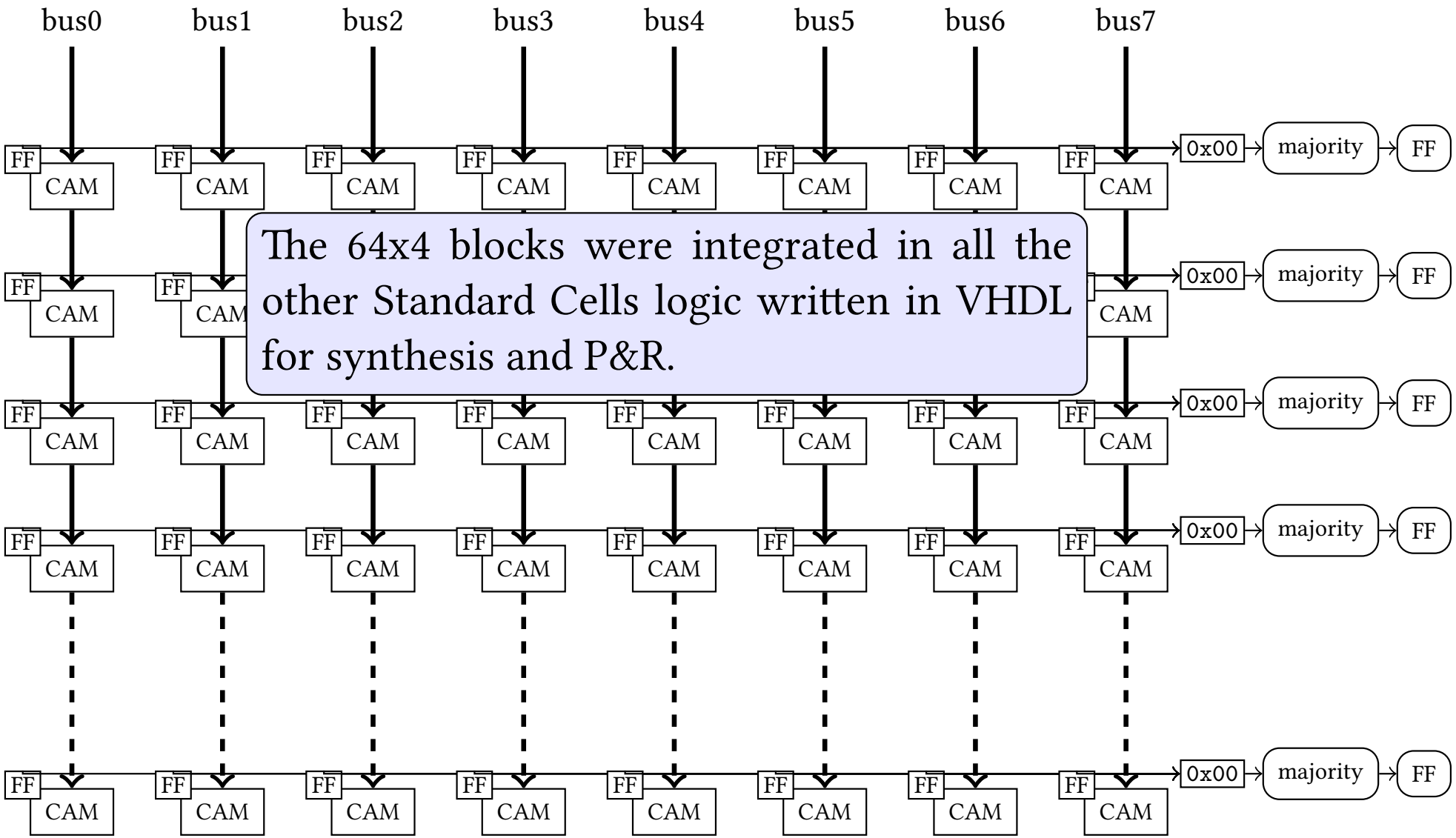
L. Frontini, S. Shojaii, A. Stabile and V. Liberali, "A new XOR-based Content Addressable Memory architecture," Electronics, Circuits and Systems (ICECS), 2012 19th IEEE International Conference on, Seville, 2012, pp. 701-704. doi: 10.1109/ICECS.2012.6463629

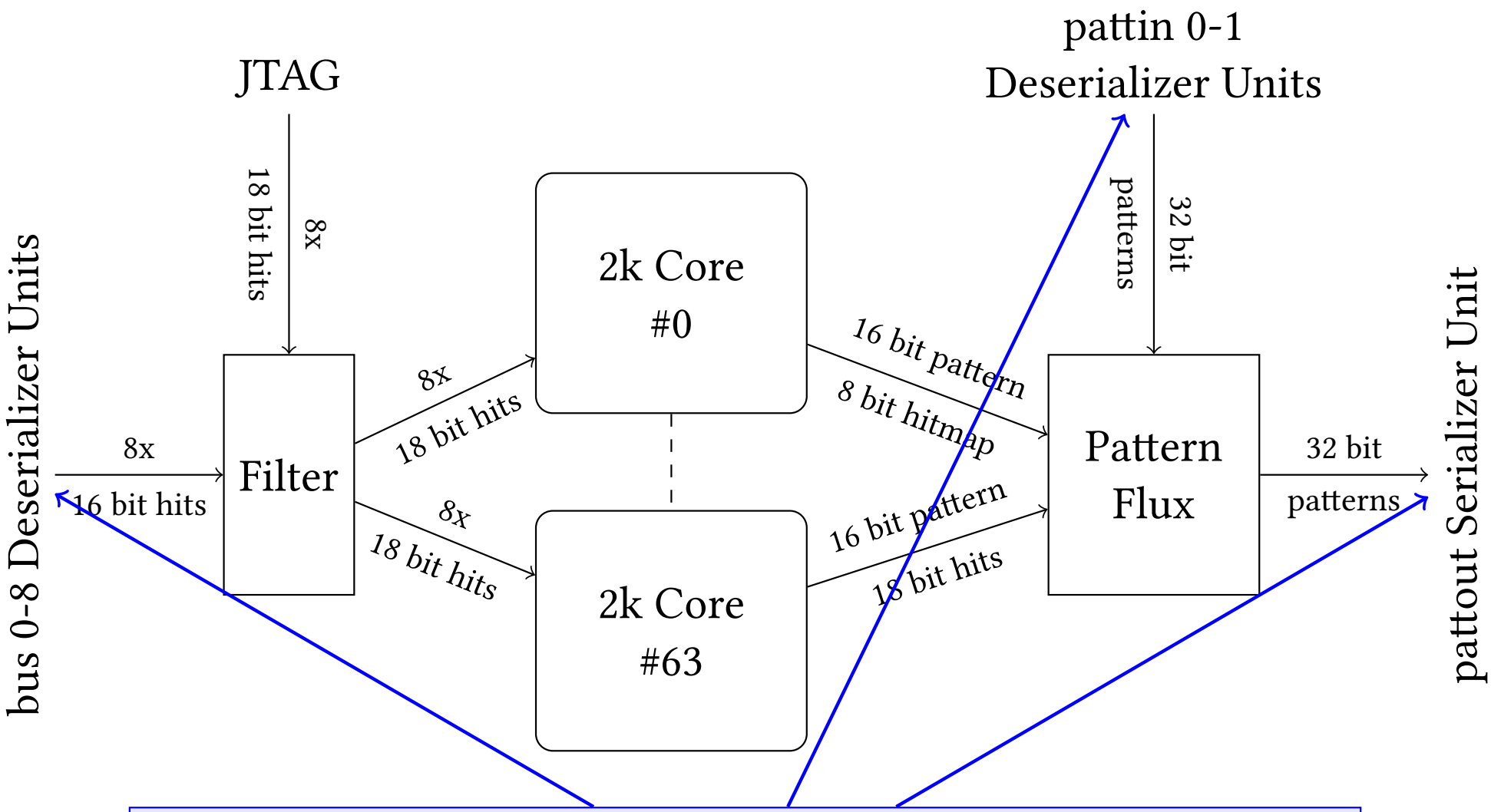
We assembled in Virtuoso a block of 64x4 CAM rows in order to optimize the area and the power consumption due to bitline propagation (major contribution to power)



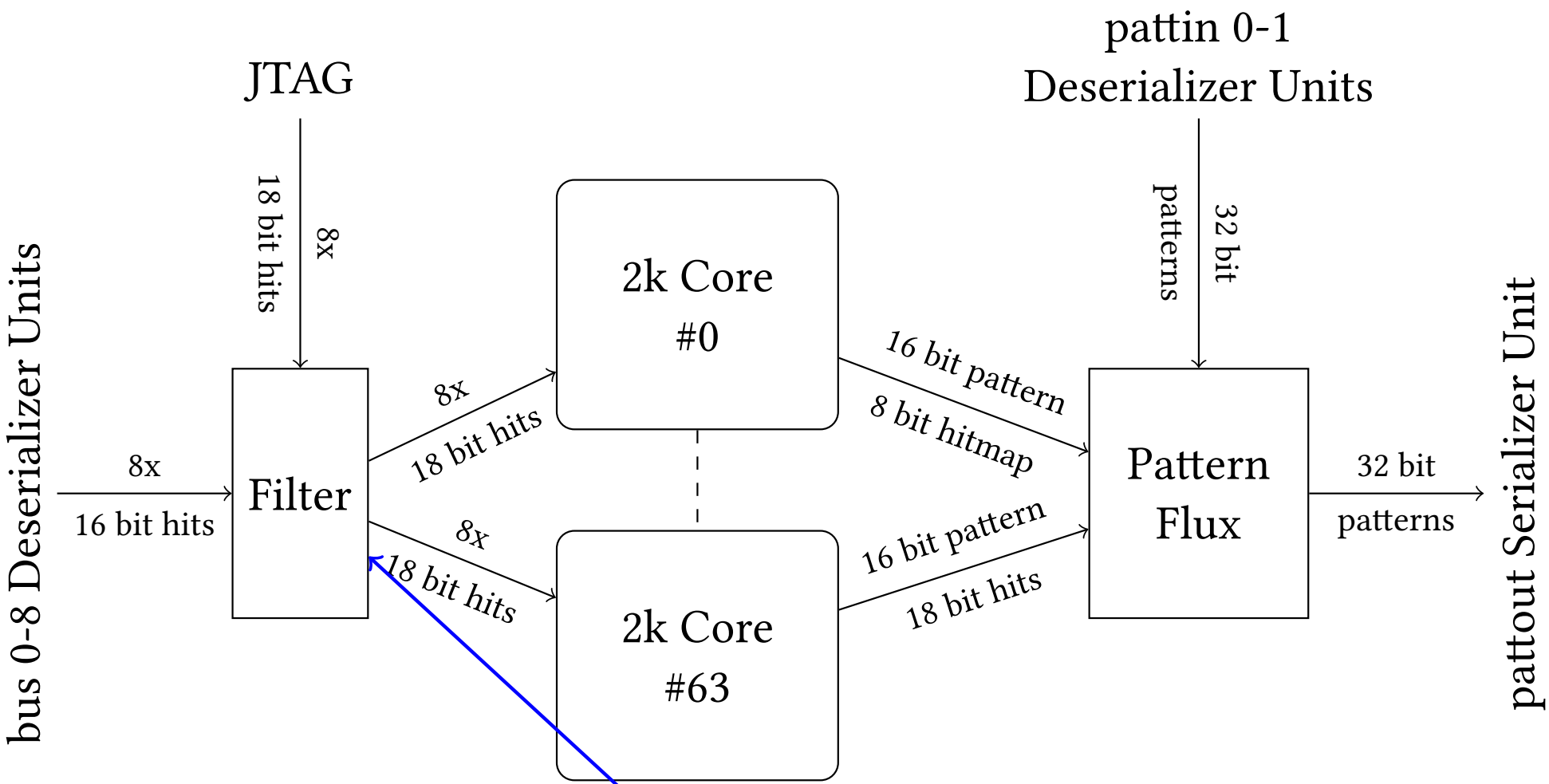


We performed simulations on the 64x4 block (full details enabled only on first, middle and last row to speedup the simulation) to extract the models for timing and power.
We also wrote a System Verilog model for behavioural simulation.

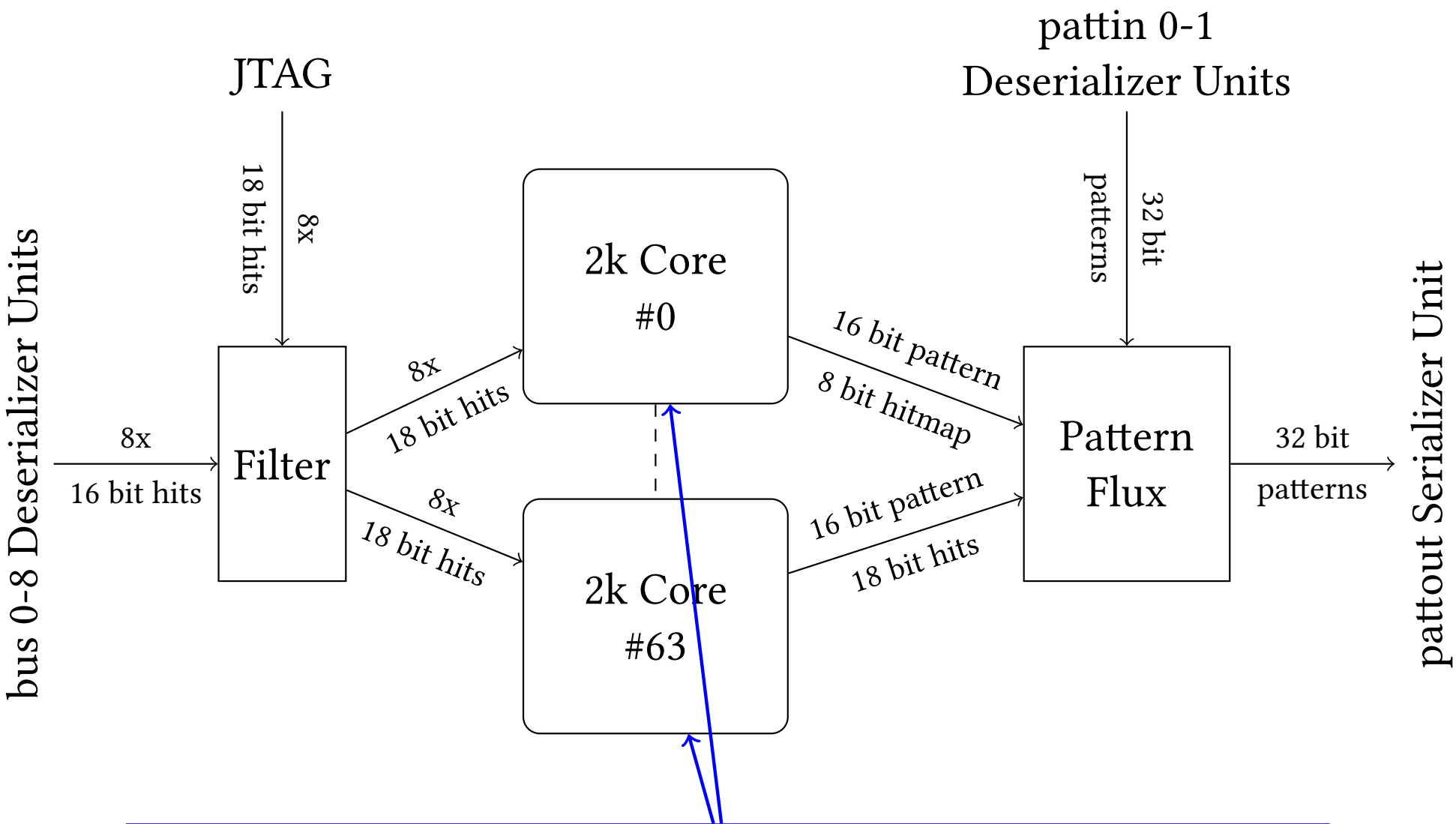

bus0
bus1
bus2
bus3
bus4
bus5
bus6
bus7
FF
CAM
0x00
majority
FF
The 64x4 blocks were integrated in all the other Standard Cells logic written in VHDL for synthesis and P&R.



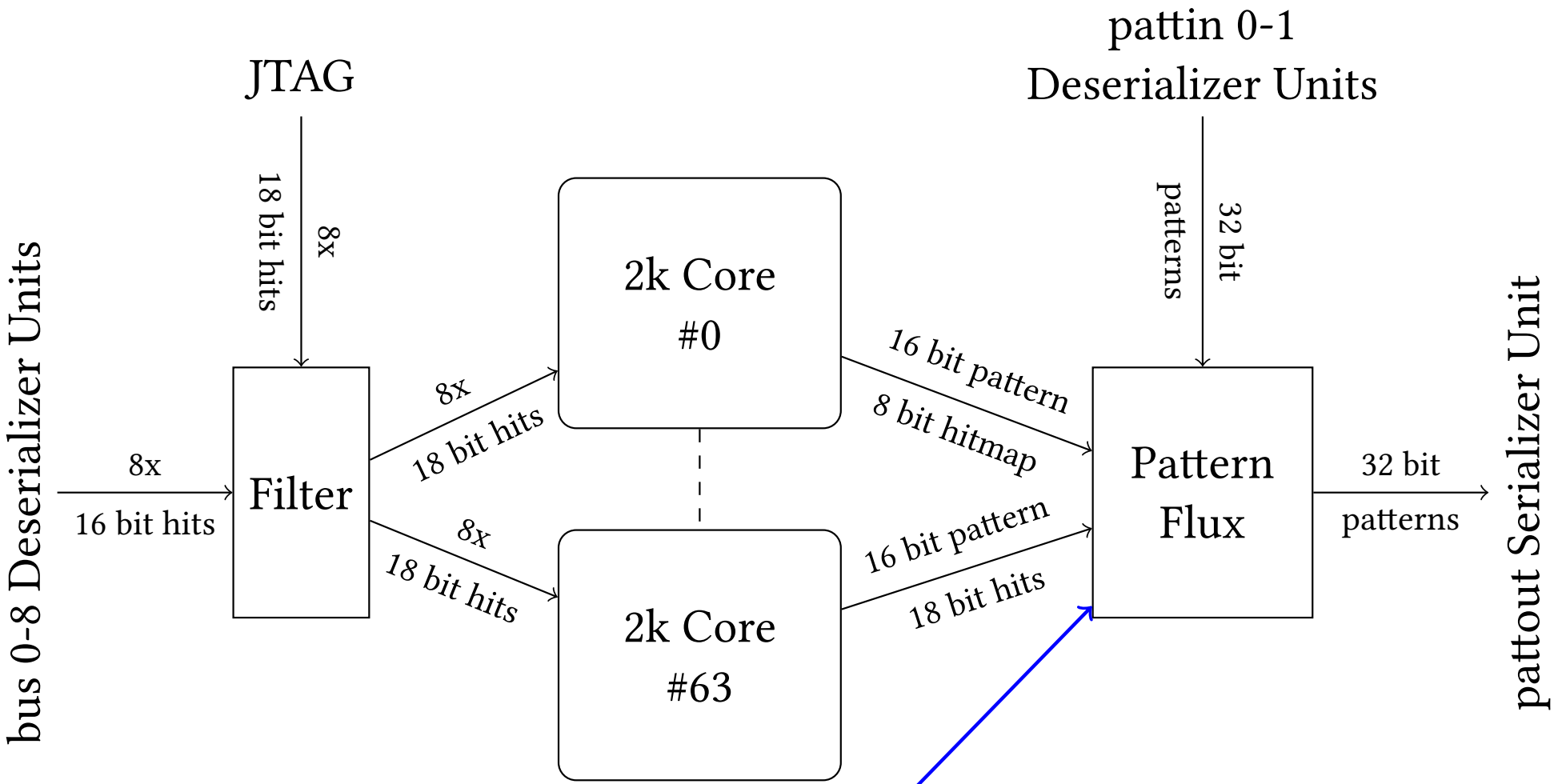
The Serializer and Deserializer units are interfaces between the serial link Hard IPs, 8b/10b decoders, data stream decoders and the main AM logic. Each Hard IP produces its own clock and a FIFO is used to cross to the main clock domains.



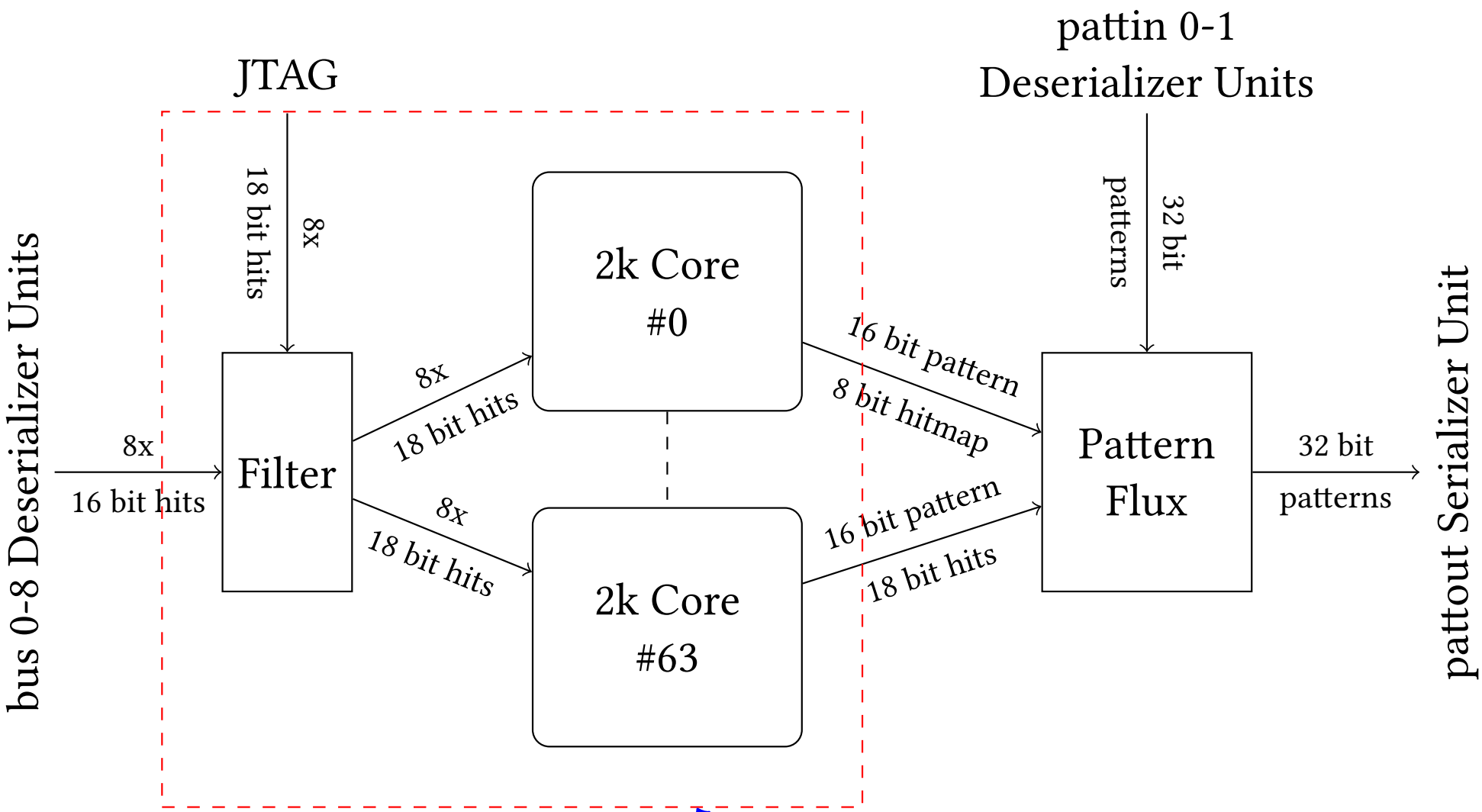
The Filter block selects between JTAG and bus data. It also encodes the incoming 16 bit bus data with configured *don't care* 18 bit format



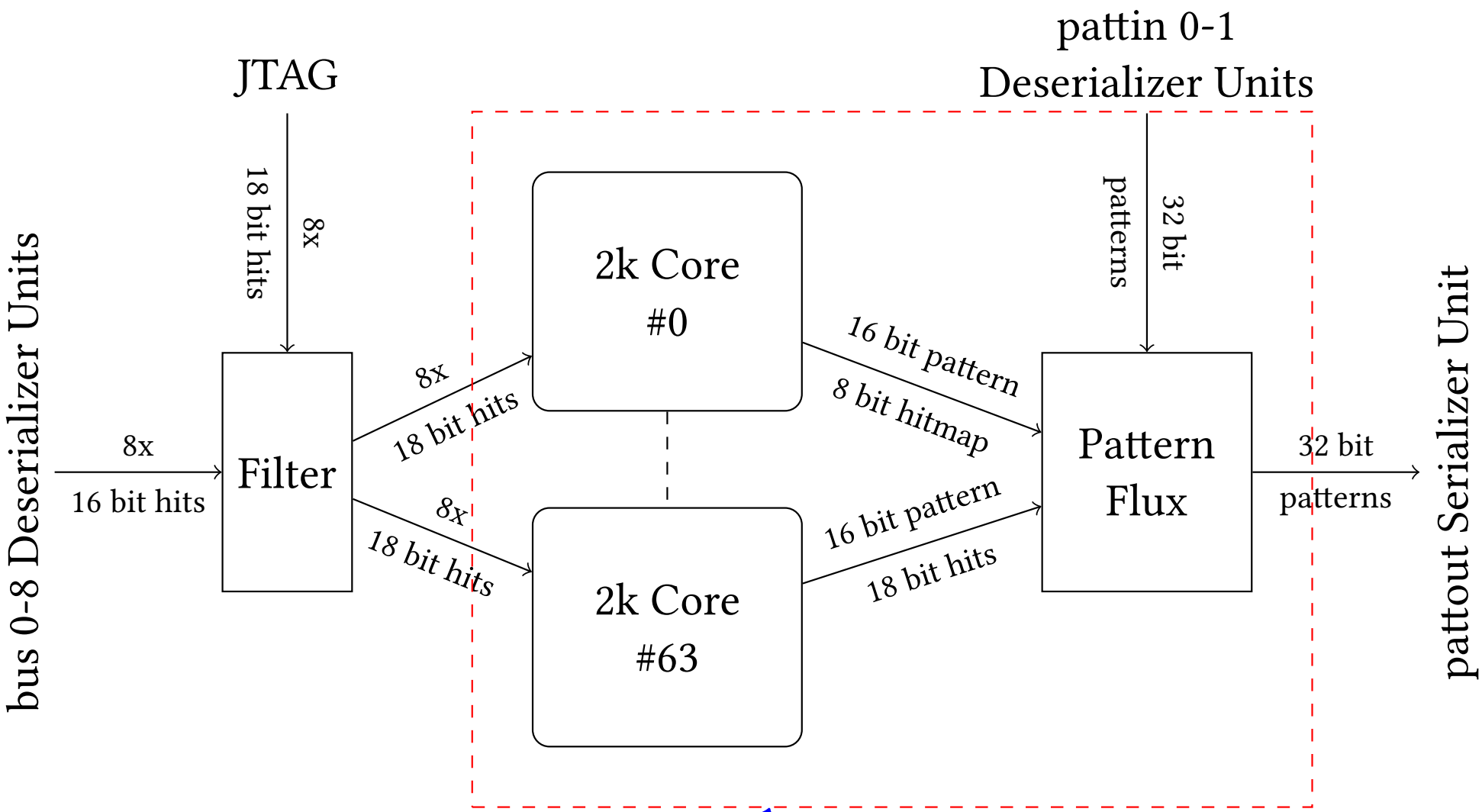
The AM is organized in 64 cores of 2 kpatt each. A block is a P&R'ed standalone and then imported in the top-level as 64 clones.



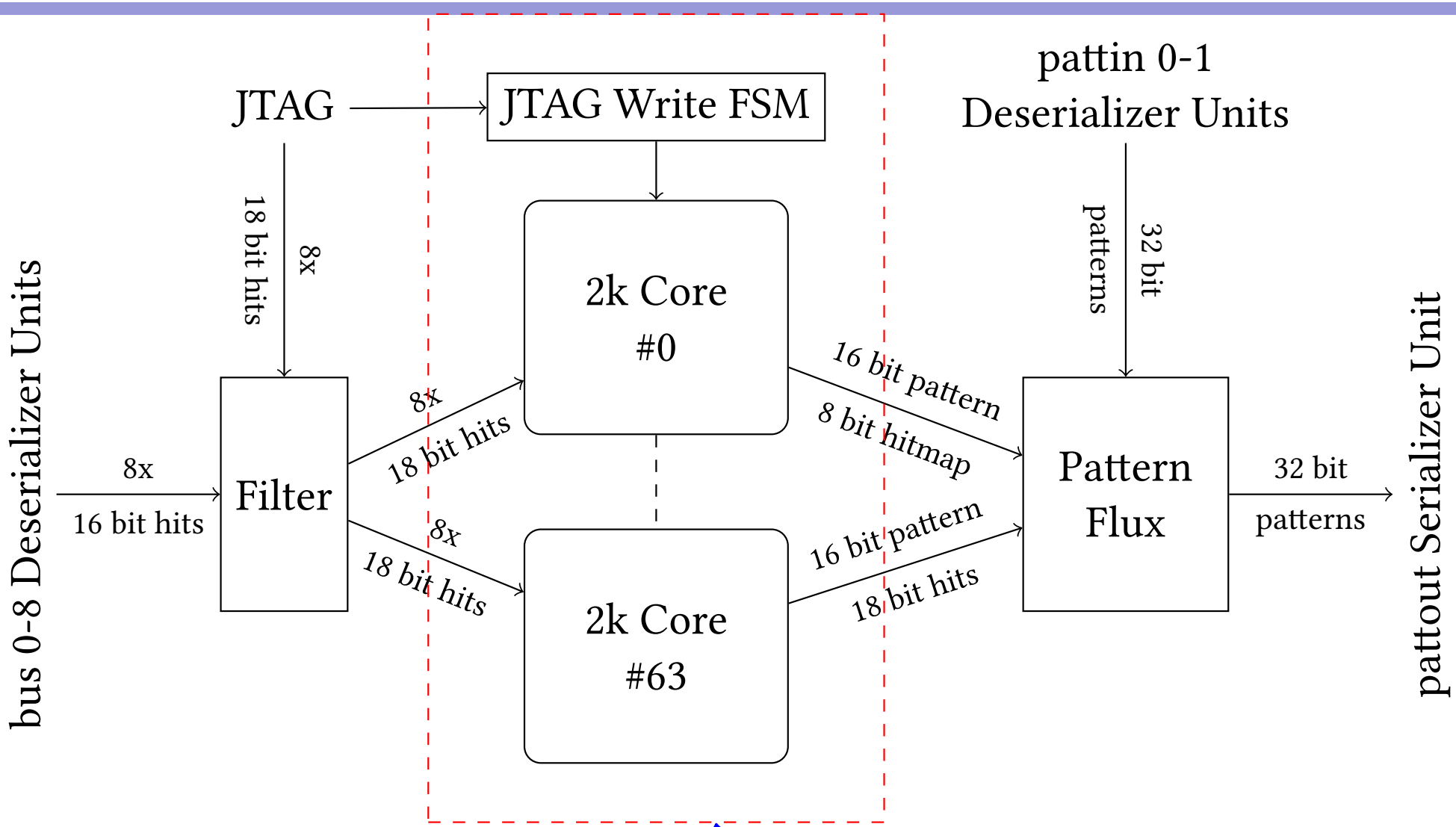
The Pattern Flux block merge all the streams of found pattern coming from the internal cores and the two external pattern inputs to the pattern output.



Data is sent to the AM for matching at CLK rate (target 100 MHz)



Matched pattern readout is done at the pattern output data rate (target 32 bit at 60 MHz)



Patterns are stored in the AM using the JTAG interface (target 40 MHz for TCK, $\sim$ 200 cycles to write one pattern)

Remarks on the flow

Flow

- Based on Foundation Flow from Cadence
- **NOT** a partitioned/hierarchical flow, but two "nested" flat flows
  - 2k block
  - Top level with 64x cloned 2k blocks
- Full flow up to (not closed) signoff took about 24h on at least 64 Gb RAM machine
- Final signoff timing closure optimizations were very hard to perform
  - Many iterations Tempus to Encounter
  - Need to run on >128 Gb RAM

AMchip06

AMchip06 specs

Technology	TSMC 65 nm
Area	$\sim 168\text{mm}^2$
Patterns	131072
Inputs (hit)	8x max 2 gbps
Inputs (patt)	2x max 2.4 gbps
Output (patt)	1x max 2.4 gbps
Core voltage	1 V to 1.2 V
Main clock	100 MHz

Conclusions

- The mixed full-custom / standard cells approach proved again to be very effective in designing large area and complex chips with area/power optimization
  - We used it since AMchip04, our first 65 nm prototype
- $\sim 168\ \mathrm{mm}^2$ proved to be very hard to handle especially at signoff timing closure
  - We lost a lot of time before gathering enough computing resources to complete the final step
- The chip works withing specifications and we are finding 84% yield for zero-defect chips (preliminary)
- We are going to use it soon in FTK!
  - Many thanks to all the supporting institutions
  - Many thanks to the microelectronics experts from CERN and the many reviews, it was a precious input!