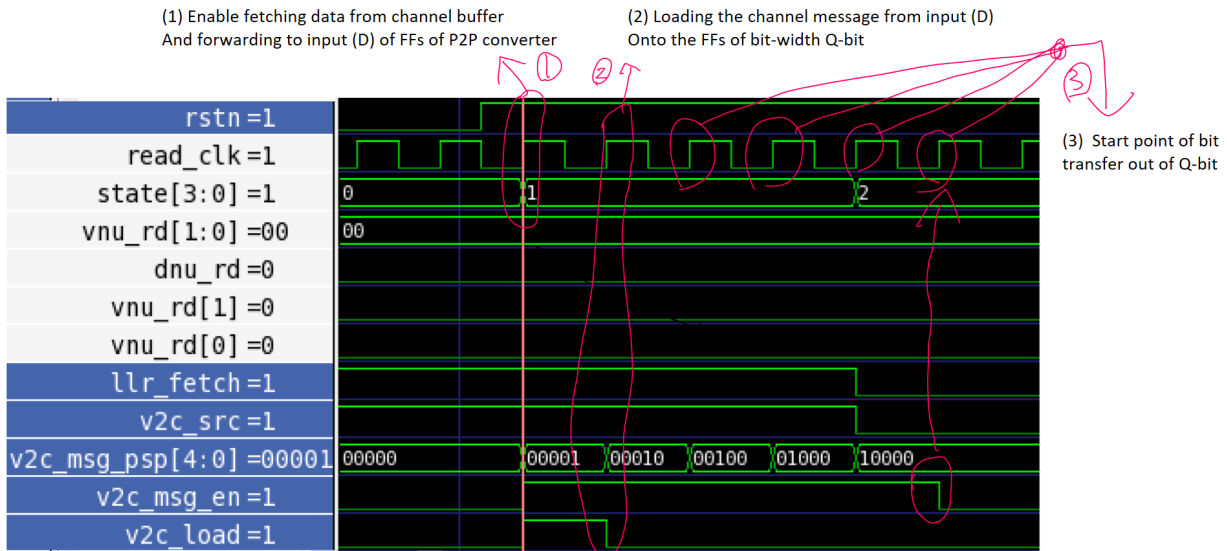


## Control Signals – Initially loading channel message and message passing to CNU

- **Signal instantiation:** llr\_fetch, v2c\_src, v2c\_msg\_en, v2c\_load
- Note
  - Although one codeword decoding process is only required fetching the channel messages from channel buffer in one clock cycle, the llr\_fetch is still asserted for four rising edges of read clock. This is because the propagation delay from channel buffer till VNUs may be so long that message passing cannot be completed in one clock cycle. Thus the additional Q-bit registers of depth 4 are inserted in between.

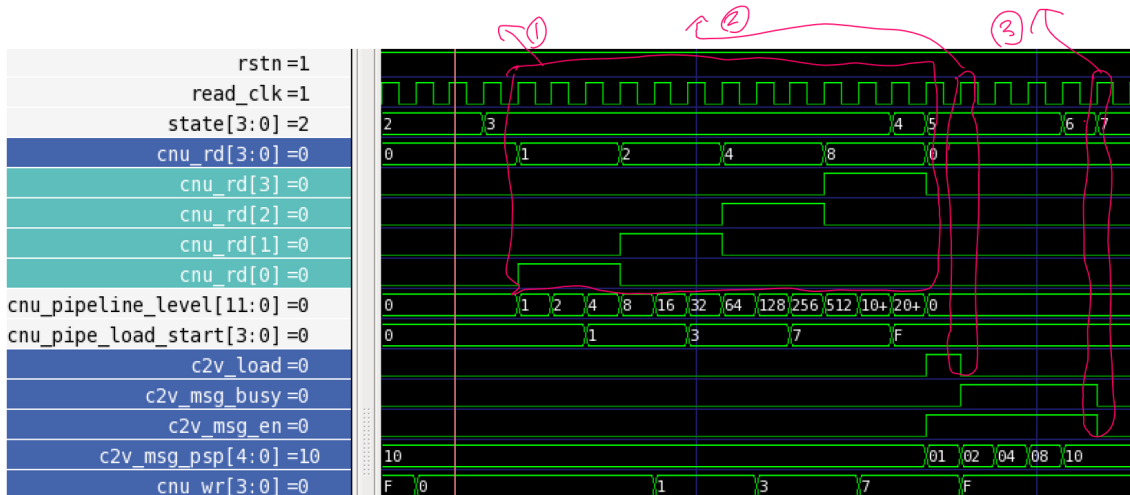


- **Signal instantiation:** cnu\_rd, c2v\_load, c2v\_msg\_busy, c2v\_msg\_en and c2v\_msg\_psp

(1) Asserting cnu\_rd to enable the check node process where the intrinsic messages are forwarded in a cascading manner.

(2) Loading the c2v messages onto the FFs of P2P converter

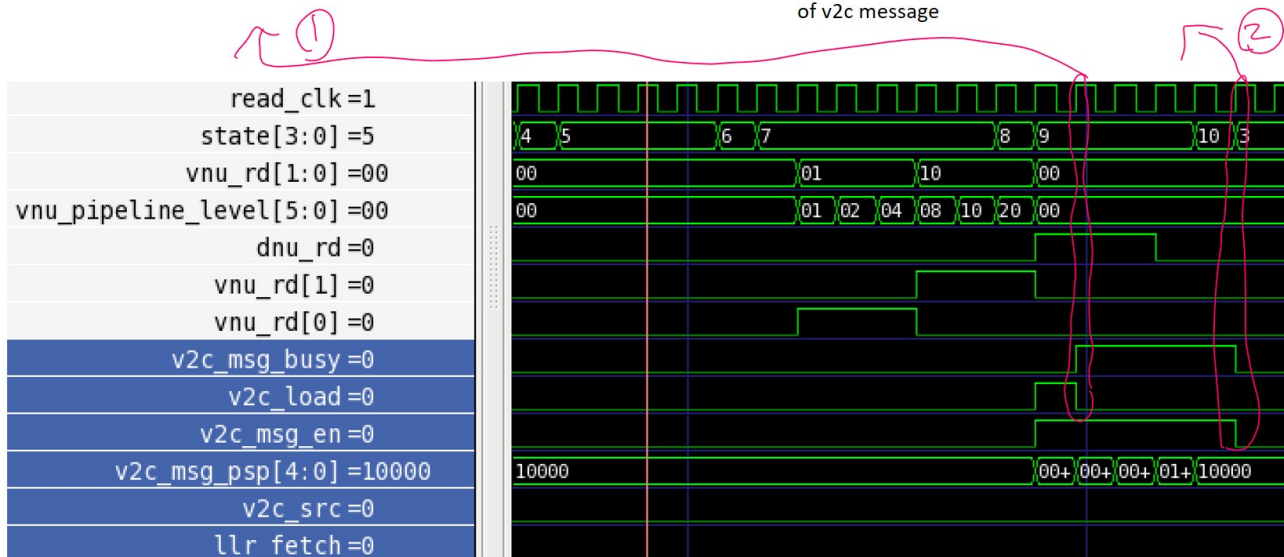
(3) Start point of last bit message passing of c2v message



- **Signal instantiation:** v2c\_msg\_busy, v2c\_load and v2c\_msg\_en

(1) Loading the v2c messages onto the FFs of P2P converter

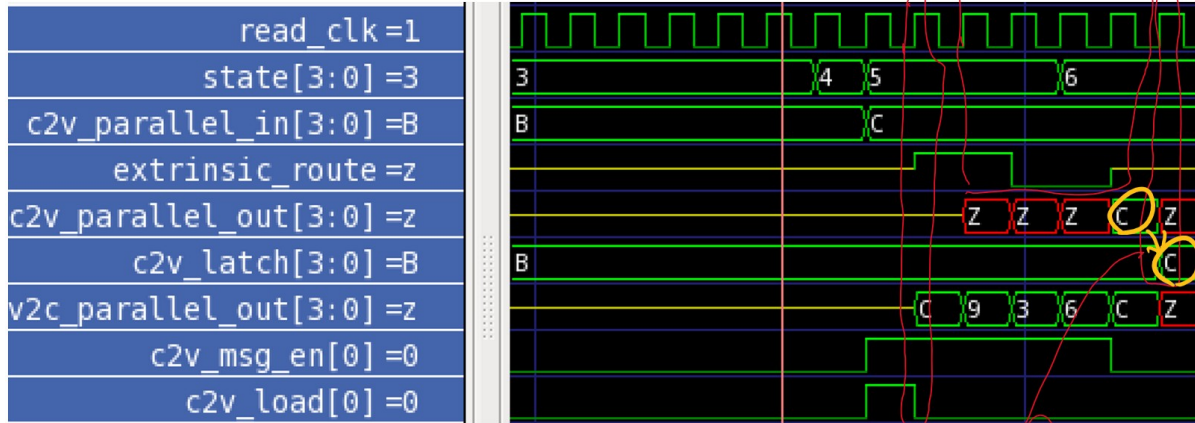
(2) Start point of last bit out of Q-bit transfer of v2c message



**Signal instantiation: c2v\_parallel\_in, extrinsic\_route, c2v\_parallel\_out, c2v\_latch, c2v\_msg\_en, c2v\_load**

(1) PSP convert latched the c2v\_load control signal and load the c2v message from last 2-LUT of CNU

(2) Transmitting c2v message in a bit-serial manner, started from MSB.



(3) Once clock cycle right after completion of transmission, the c2v buffer latches the received c2v parallel message which will be fetched by VNU. That is, c2v\_latch latches the data from c2v\_parallel\_out