Concatenated FEC Codes for 800GE and 1.6TE

Zhongfeng Wang
Nanjing Univ.

Chuan Zhang
Southeast Univ.

IEEE802.3dj (Online), Feb. 2023

Outline

- KR FEC for 100GE
- KP FEC for 400GE
- Concatenated FEC Codes for Hi-Speed Networking
- KP FEC Based Concatenated Codes for Ethernet
- Simple Analysis about Candidate FEC Codes
- Conclusion

KR- FEC Code for 100GE

- KR-FEC
  - □ Transcoding: 4×64b/66b => 256/257bit coding (264 => 257)
  - □ FEC: RS(528, 514, t = 7), processing latency < 100ns
  - □ 100GBASE-KR4/CR4/SR4, 25GBASE-R
  - □ First proposed in IEEE 802.3bj meeting in Nov. 2011*

- Good Properties
  - □ Net OH: 0% (257/264 * 528/514 = 1)
  - □ **No dummy or padding bits, good for coding gain and hardware cost**
  - □ Good parallelism options: 5280/33 = 2*80bits = 160bits
    - ◆ 644.53125M*160 = 103.125G, 625M*160 = 100G, 625M = 4*156.25
    - ◆ For 4 physical lanes, can distribute/receive 4 RS symbols to/from each lane each clock cycle
    - ◆ Can finish encoding or decoding of one FEC block within an integer number of cycles

* Z. Wang, et al, "Further Studies of FEC Codes for 100G-KR", Atlanta, Nov. 2011

KP- FEC Code for 200/400GE

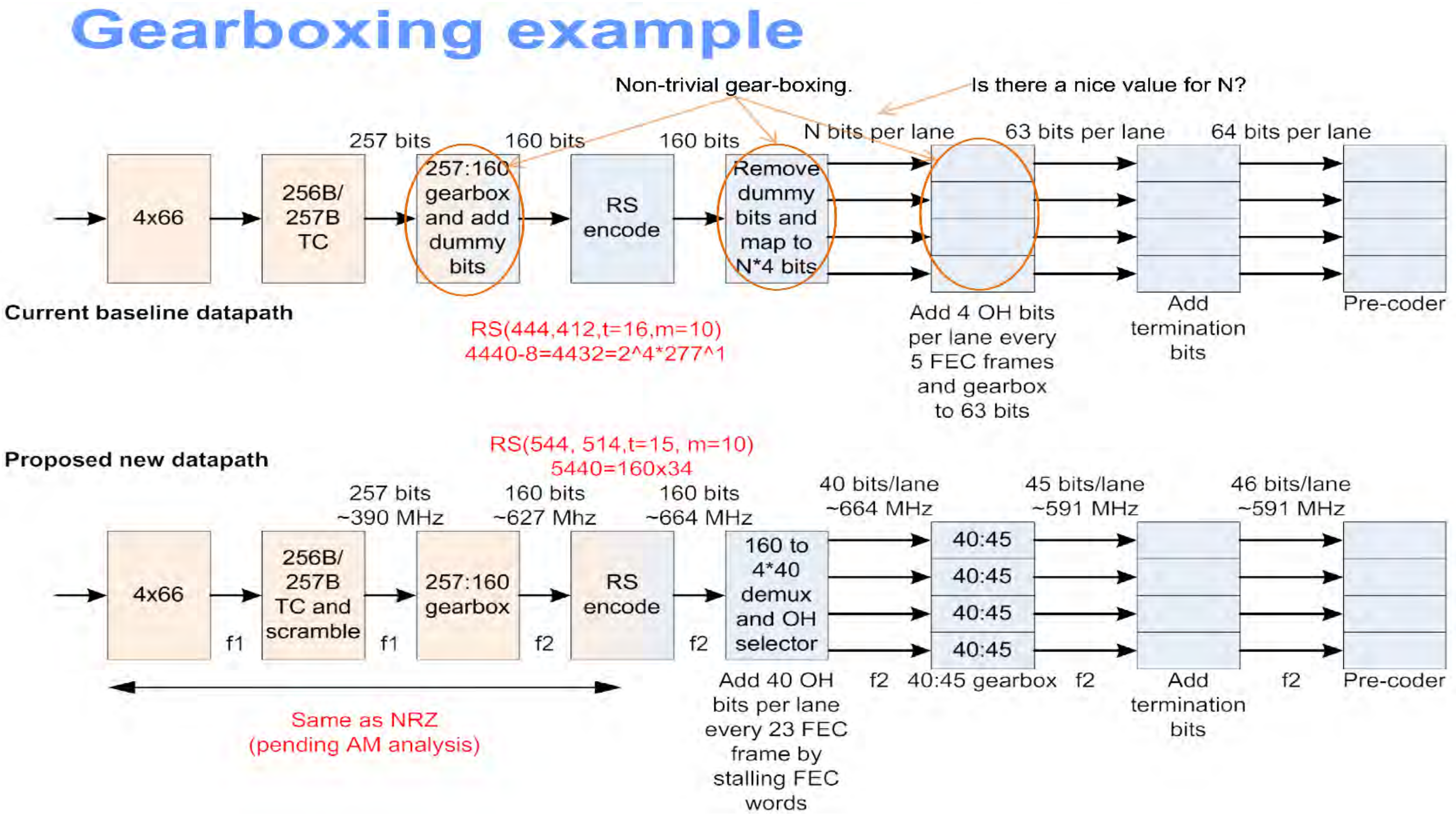
- KP-FEC
  - □ RS(544, 514, t = 15), processing latency ~ 100ns
  - □ 100GBASE-KP4, 400GE, 200GE, and 50GE
  - □ First proposed at IEEE 802.3bj meeting in May 2012*

- Good Properties
  - □ **No dummy or padding bits, good for coding gain and hardware cost**
  - □ Parallelism options: 160/320, can use same bus-width as KR-FEC: 5440/34 = 160bits
  - □ Net OH (ensure integer PLL, 680*156.25M = 106.25G) (68/66 – 1) ~ = 3%

* M. Brown and Z. Wang, "100G Backplane PAM4 PHY FEC/PMA Encoding Enhancements", Minneapolis, May 2012

FEC Consensus in 802.3bj 100G PAM4 PHY

- **Proposal of RS(544,514) to Replace RS(444,412)**
  - □ Transcoded PCS data fits exactly into payload.
    - ◆ No gearboxing for dummy bits.
  - □ More efficient use of parity.
    - ◆ Permits reference clock factor reductions from 88 to 87

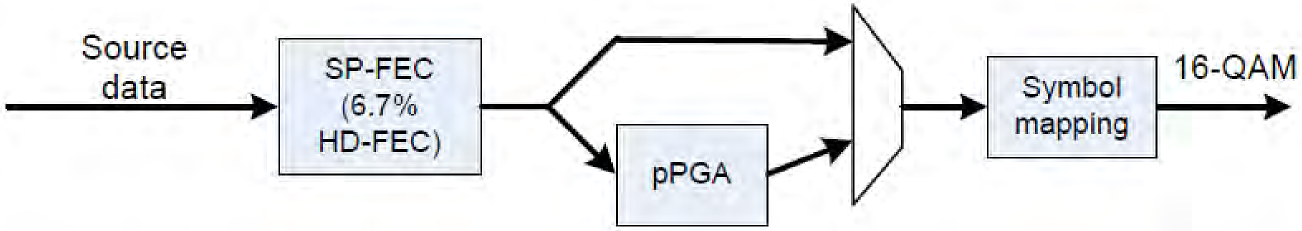


- **Broad consensus and adopted in IEEE 802.3bj meeting in May 2012**

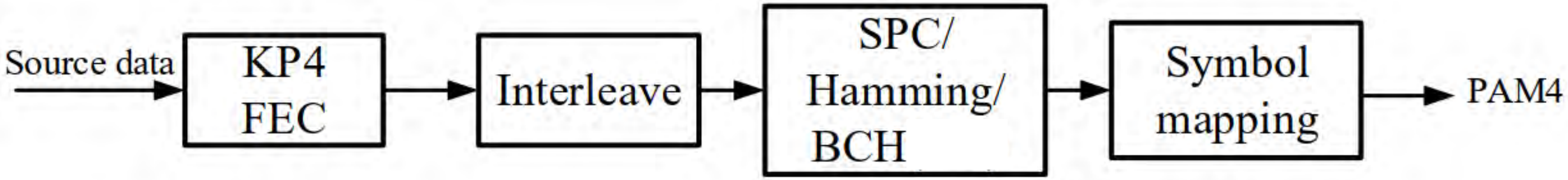
* M. Brown and Z. Wang, "100G Backplane PAM4 PHY FEC/PMA Encoding Enhancements", Minneapolis, May 2012

Concatenated Codes for Networking

- Applications:
  - Boost coding gain of existing networking systems
    - Proposed (*) the framework to adjust overall coding gain via code concatenation at ITU Q11/15 meeting in March 2014



  - Obtain an optimal tradeoff between coding gain, power and hardware complexity
    - KP-FEC concatenated with SPC/Hamming/BCH code can achieve a NCG of more than 8.5 dB



* Z. Wang, "Super-Product BCH(SP-BCH) Code for 100G and Beyond Networking ", Mar.-Apr., 2014

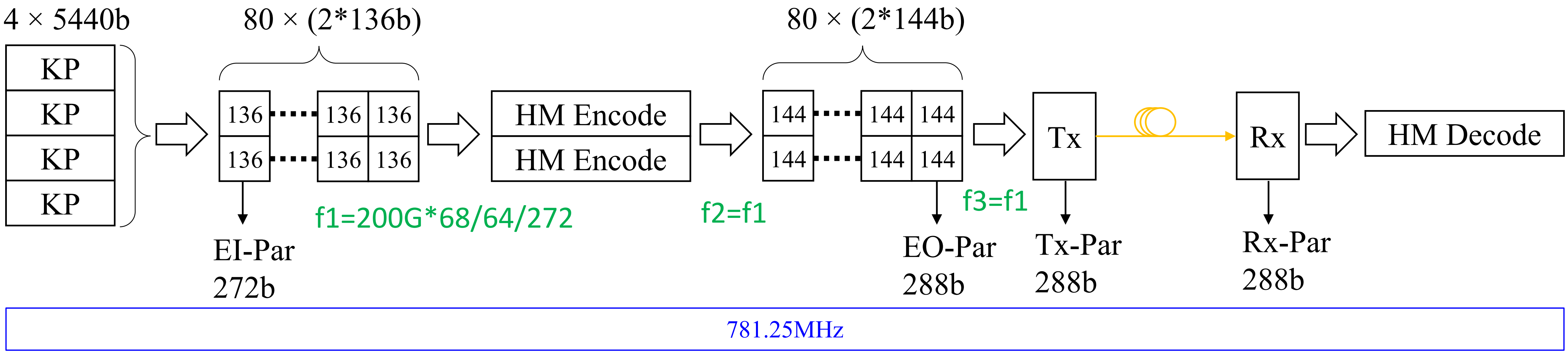
Candidate FEC Code (1)

- KP-FEC × N + HM(144,136) × M
  - No dummy bits, integer PLL
  - Code rate: 1/(68/66 * 18/17) = 66/72 = 11/12
  - OH: 72/66 – 1 = **9.09 %**

- Parallel Processing Considerations
  - 4×5440 = **136*160** = 272 * 80, assume 4×interleaved KP-FEC as outer code
    - 200G data rate: clk < 1.0GHz, EI-Par = 136*2, EO-Par = Tx-Par = Rx-Par = 144*2,
    - 400G data rate: double 200G parallelism
    - 800G data rate: double 400G parallelism
    - 1.6T data rate: double 800G parallelism

* EI-Par: encoder input side parallelism, EO-Par: encoder output side parallelism, Tx-Par: data out parallelism

Candidate FEC Code (1) Function Illustration



- Good Properties:
  - □ Easy parallel processing for both Tx and Rx
  - □ Simple control logic (integer cycles to process a composite FEC block)
  - □ **No gearbox is needed (no data rotation/wrap up, et al)**
  - □ **No additional latency by gearbox translation**
  - □ **Very simple clocking**

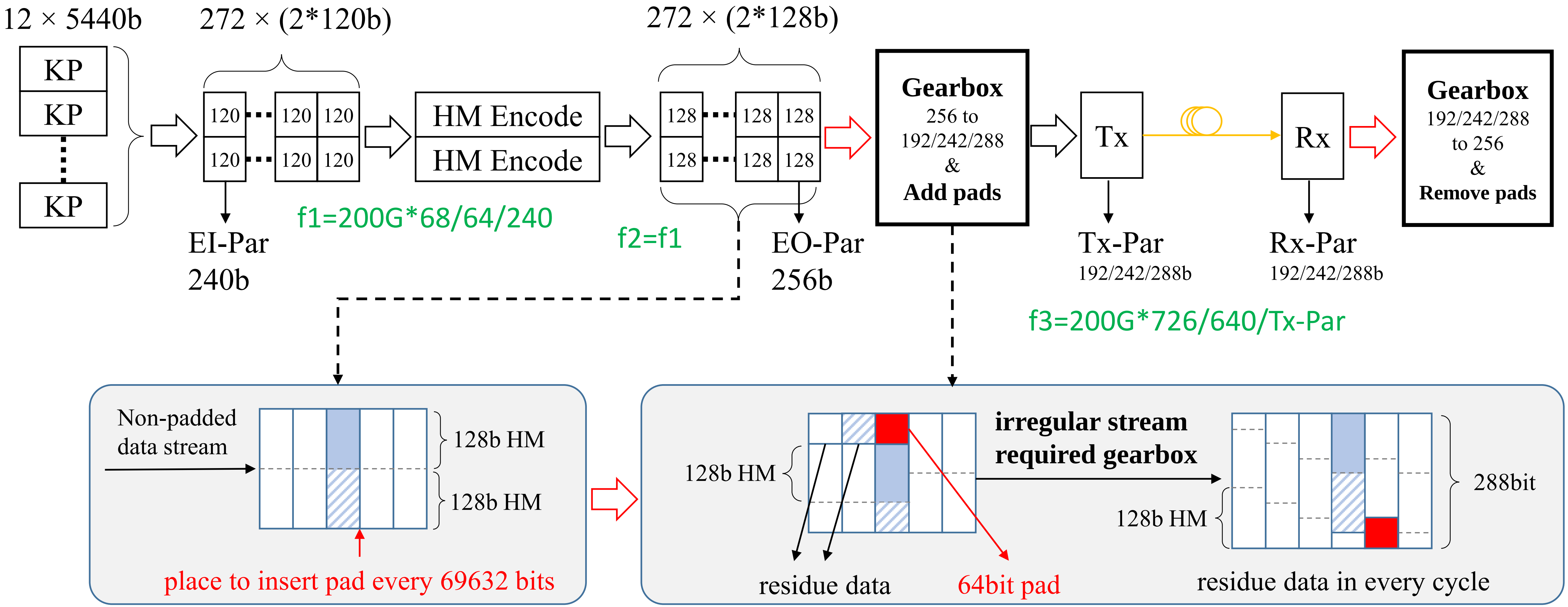
Candidate FEC Code (2)

- KP-FEC × N + HM(128,120) × L + Padding bits × D
  - 69632 (= 128*544) + 64 bits (padding)
  - Code rate: 1/[68/66*128/120 *(69632+64)/69632] = 660/726
  - OH: 726/660 – 1 = **10.0 %**
- Parallel Processing Considerations:
  - 69632 + 64 = 128*544 + 64
    - 200G data rate: (69632+64) = 363*3*64, RxPar = 192/242/288
    - 200G data rate: encoder input side, 120*544 = 240*272; EI-Par = 240

      encoder output side， **EO-Par = 256** (using same clock)

      TxPar = 192/242/288; need gearbox， costing at least 1 extra cycle at each direction.
    - 400G/800G/1.6T data rate: can simply 2x/4x/8x parallelism of 200G case
    - If we don’t want gearbox between encoder output and Tx, we can consider EO-Par = EI-Par = TxPar. Encoder circuitry (including Fclk) will be more complicated since each coder gets irregular stream.
  - 69632*2 + 128 = 128*544*2 + 128
    - Similar parallel processing options to above can be considered
- Good Properties:
  - Dummy bits can be used for data synchronization. However, it is not trivial to design a data synchronization scheme because padding bits have around 0.4% of error rate.

Candidate FEC Code (2) Function Illustration



12 × 5440b

KP
KP
KP

272 × (2*120b)

120 120 120
120 120 120

EI-Par
240b

f1=200G*68/64/240

HM Encode
HM Encode

f2=f1

272 × (2*128b)

128 128 128
128 128 128

EO-Par
256b

Gearbox
256 to
192/242/288
&
Add pads

Tx

Tx-Par
192/242/288b

Rx

Rx-Par
192/242/288b

Gearbox
192/242/288
to 256
&
Remove pads

f3=200G*726/640/Tx-Par

Non-padded
data stream

128b HM
128b HM

place to insert pad every 69632 bits

128b HM

residue data

64bit pad

irregular stream
required gearbox

128b HM

288bit

residue data in every cycle

Summary about Two Candidate Codes

Inner code	NCG	Over clocking	Codec complexity	Design & verification	Total Rx power increase	Gearbox Latency
(144,136)	very close	9.09 %	Lower	Easier	0% (ref. design)	None
(128,120) +padding	very close	10.0 %	Higher	Complex	15~20% x power of (ref-codec)	+ 2 cycles Tx+Rx at least

- In principle, we can also add dummy bits periodically for the first coding scheme, e.g.,
  - for every 103680 (=720*144) bits, add 144bits, 1/code rate = 721/660 < 726/660
  - for every 103680 (=144*720) bits, add 288bits, 1/code rate = 722/660 < 726/660
- Added dummy bits can be used for data alignment. But it is non-trivial to use them to convey control bits due to considerable raw error rate of them.
- Considering the pros and cons, neither of the schemes that add dummy bits is promising.

Final Remarks

- KR FEC and KP FEC were Chosen for 100GE and 400GE Respectively, but for Some Common Reasons:
  - □ No dummy bits are involved and can ensure integer PLL
  - □ Easy for parallel processing
  - □ Low-complexity implementation, facilitating low latency and low power design

- Existing Two Candidate Inner Codes Have almost Same Performance (NCG). Thus Easy Implementation/Low Cost/Low Power/Low Latency Should be Deeply Considered.

- We Should Work Together to Make Our Industry "Wear Shorts to Race 100 Meters"

Thanks for Your Attention!

Email： zfwang at NJU.edu.cn