



- (51) **International Patent Classification:**
H03M 13/11 (2006.01)
- (21) **International Application Number:**
PCT/US2017/053128
- (22) **International Filing Date:**
23 September 2017 (23.09.2017)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
62/416,584 02 November 2016 (02.11.2016) US
15/712,845 22 September 2017 (22.09.2017) US
- (71) **Applicant: QUALCOMM INCORPORATED** [US/US];
ATTN: International IP Administration, 5775 Morehouse
Drive, San Diego, California 92121-1714 (US).
- (72) **Inventors: LONCKE, Vincent;** 5775 Morehouse Drive,
San Diego, California 92121-1714 (US). **VARATKAR,**
Girish; 5775 Morehouse Drive, San Diego, California
92121-1714 (US). **RICHARDSON, Thomas Joseph;** 5775
Morehouse Drive, San Diego, California 92121-1714 (US).
CAO, Yi; 5775 Morehouse Drive, San Diego, California
92121-1714 (US).
- (74) **Agent: READ, Randol W.** et al.; Patterson & Sheridan,
L.L.P., 24 Greenway Plaza, Suite 1600, Houston, Texas
77046 (US).
- (81) **Designated States** (*unless otherwise indicated, for every
kind of national protection available*): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN,
HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP,
KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME,

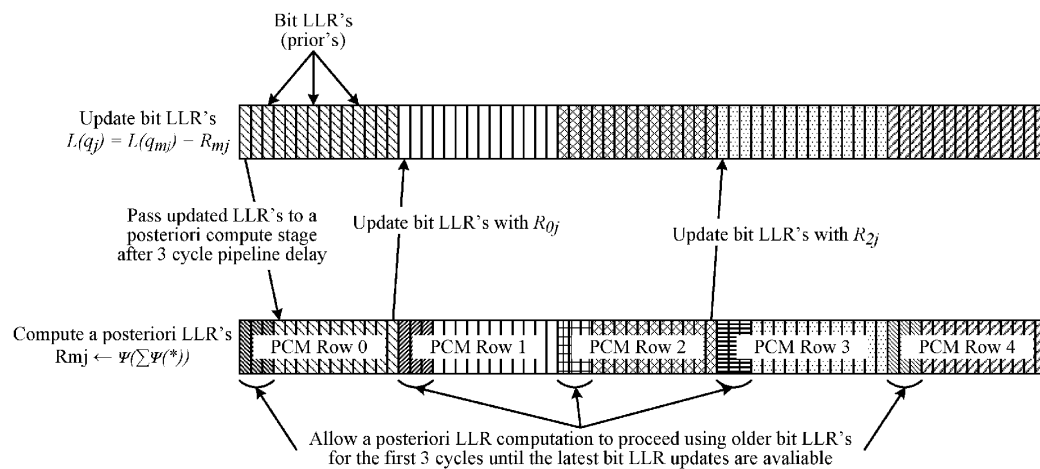
(54) **Title:** PIPELINED HIGH-THROUGHPUT LAYERED LDPC DECODER ARCHITECTURE

FIG. 13

(57) **Abstract:** Certain aspects of the present disclosure generally relate to methods and apparatus for decoding low density parity check (LDPC) codes, and more particularly to a deeply-pipelined layered LDPC decoder architecture for high decoding throughputs solving memory consistency and conflict problems. Aspects of the present disclosure present techniques for mitigating pipeline delays, for example, by relaxing the dependency between updating bit LLRs and computing check node messages, so that for a particular row, the check node processing can use the latest available bit LLR's (e.g., outdated bit LLRs) rather than waiting for the latest updates (e.g., updated bit LLRs) to occur. Memory consistency conflicts are avoided by storing the latest available bit LLRs in a LLR memory and updating the bit LLRs with the difference between old and new check node messages. Furthermore, logically splitting the LLR memory into dual banks enables the decoder to read or write from both banks of memory at the same time, thereby increasing the read/write bandwidth. Advantageously choosing the parity check matrix row (PCM) calculation order, sorting the check node messages and the bit

MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ,
OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,
SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

LLR updates in memory, and/or choosing a memory bank to store the check node messages and bit LLR updates based on, for example, dependencies between rows in the PCM alleviates memory conflicts/consistency errors and reduces pipeline processing delays.