

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2023/0231573 A1 SFORZIN et al.

(43) **Pub. Date:**

Jul. 20, 2023

(54) ITERATIVE ERROR CORRECTION IN MEMORY SYSTEMS

- (71) Applicant: Micron Technology, Inc., Boise, ID
- Inventors: Marco SFORZIN, Boise, ID (US); DI HSIEN NGU, Boise, ID (US)
- Assignee: Micron Technology, Inc., Boise, ID (73)(US)
- (21) Appl. No.: 17/843,171
- (22) Filed: Jun. 17, 2022

Related U.S. Application Data

(60) Provisional application No. 63/301,028, filed on Jan. 19, 2022.

Publication Classification

(51) Int. Cl. H03M 13/09 (2006.01)H03M 13/11 (2006.01)H03K 19/173 (2006.01) (52) U.S. Cl.

CPC H03M 13/098 (2013.01); H03M 13/1174 (2013.01); H03M 13/1171 (2013.01); H03K 19/1737 (2013.01)

(57)ABSTRACT

A system and method for detecting and correcting memory errors in CXL components is presented. The method includes receiving, into a decoder, a memory transfer block (MTB), wherein the MTB comprises data and parity information, wherein the MTB is arranged in a first dimension and a second dimension. An error checking and a correction function on the MTB is performed using a binary hamming code logic within the decoder in the first dimension. An error checking and a correction function on the MTB is performed using a non-binary hamming code logic within the decoder in the second dimension. Further, the binary hamming code logic and the non-binary hamming code logic perform the error checking on the MTB simultaneously.

1500 1505 receiving, into a decoder, a memory transfer block (MTB), wherein the MTB comprises data and parity information. wherein the MTB is arranged in a first dimension and a second dimension -1510performing an error checking and a correction function on the MTB using a binary hamming code logic within the decoder in the first dimension -1515performing an error checking and a correction function on the MTB using a non-binary hamming code logic within the decoder in the second dimension; and -1520wherein the binary hamming code logic and the non-binary hamming code logic perform the error checking on the MTB simultaneously -1525generating a first codeword based on a vertical partition of the MTB using the binary hamming code logic and generating a second codeword based on a horizontal partition of the MTB using the non-binary hamming code logic determining, upon initial receipt of the MTB if either the binary hamming code logic or the non-binary hamming code logic, that the MTB contains one or more initial errors, performs an initial decoding comprising; 1538 forwarding the MTB to the binary hamming code logic for correction; outputting a MTB' from the binary hamming code logic; forwarding the MTB' to the non-binary hamming code logic for correction; and outputting a MTB" from the non-binary hamming code logic determining, after the MTB" is output from the non-binary hamming code logic, if either the binary hamming code logic or the non-binary hamming code logic determines that the MTB" still contains a remaining or a new error, performing an iterative decoding comprising: 1535 directing the MTB" back to the binary hamming code logic for correction; outputting a MTB" from the binary hamming code logic;

> forwarding the MTB" to the non-binary hamming code logic for correction; outputting a MTB" from the non-binary hamming code logic to a multiplexor; and outputting the MTB"" from the multiplexor to a CXL component