



- (51) **International Patent Classification:**  
G06F 12/08 (2006.01) G06F 11/10 (2006.01)  
G06F 13/14 (2006.01)
- (21) **International Application Number:**  
PCT/US2012/031095
- (22) **International Filing Date:**  
29 March 2012 (29.03.2012)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**  
13/078,704 1 April 2011 (01.04.2011) US
- (71) **Applicant (for all designated States except US):** INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, M/S: RNB-4-150, Santa Clara, CA 95052 (US).
- (72) **Inventors; and**
- (75) **Inventors/Applicants (for US only):** MCGINNIS, Darrell, S. [US/US]; 860 N 28th Ave, Cornelius, OR 97113 (US). HUDDLESTON, C. Scott [US/US]; 13850 SW Harness Lane, Beaverton, OR 97008 (US). AGARWAL, Rajat [US/US]; 16548 NW Talkingstick Way, Beaverton, OR 97006 (US). CHINTHAMANI, Meenakshisundaram R. [US/US]; 2616 NE Nova Avenue, Hillsboro, OR 97124 (US).

- (74) **Agents:** VINCENT, Lester J. et al.; Blakely, Sokoloff, Taylor & Zafman LLP, 1279 Oakmead Parkway, Sunnyvale, CA 94085-4040 (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, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, 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, ML, MR, NE, SN, TD, TG).

**Published:**  
— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) **Title:** MECHANISMS AND TECHNIQUES FOR PROVIDING CACHE TAGS IN DYNAMIC RANDOM ACCESS MEMORY

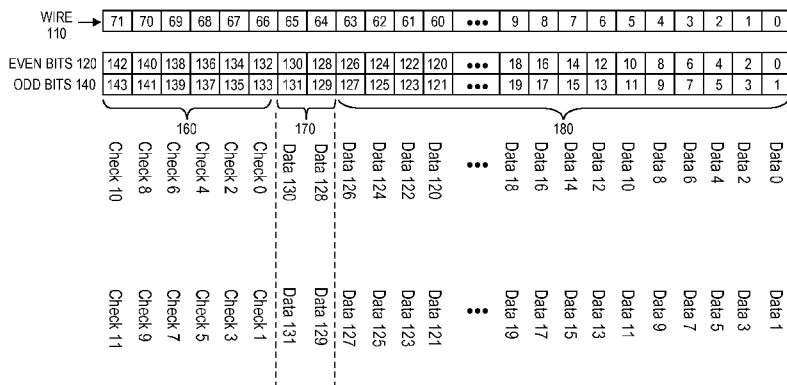


FIG. 1

(57) **Abstract:** A dynamic random access memory (DRAM) is operated as a cache memory coupled with a processor core. A block of data is transmitted to the DRAM as even and odd pairs of bits from the processor core. The block of data includes N error correct-ing code (ECC) bits and 11\*N data bits. Two or more cache lines are to be stored in a memory page with tag bits aggregated together within the page.



MECHANISMS AND TECHNIQUES FOR PROVIDING CACHE TAGS IN DYNAMIC  
RANDOM ACCESS MEMORY

TECHNICAL FIELD

Embodiments of the invention relate to use of dynamic random access memory (DRAM) as cache memory. More specifically, embodiments of the invention relate to techniques for providing cache tags in standard DRAM that is used as a cache memory.

BACKGROUND

The general use of cache memory systems has been known in the computing technologies for years. Current cache memory systems are typically multi-level cache systems in which the different cache levels are different sizes and are physically located in different places. In some larger systems dynamic random access memory (DRAM) may be utilized as at least one level of the cache memory system.

One challenge associated with cache memory structures is that information associated with a tag line, for example, tags, must also be stored. For dedicated hardware cache memory structures, this associated information may be stored separately from the corresponding cache lines. However, this strategy may prevent standard DRAM modules from being used in cache memory systems.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the invention are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings in which like reference numerals refer to similar elements.

**Figure 1** is a conceptual illustration of one embodiment of a data arrangement that may be utilized for transmitting cache lines of data to and from a physical memory device.

**Figure 2** is a conceptual diagram of one memory organization that may be utilized with the data configuration of Figure 1.

**Figure 3** is a block diagram of one embodiment of an electronic system.

DETAILED DESCRIPTION

In the following description, numerous specific details are set forth. However, embodiments of the invention may be practiced without these specific details. In other instances,

well-known circuits, structures and techniques have not been shown in detail in order not to obscure the understanding of this description.

Caches traditionally require separate or segregated storage for cache tags. Separate tag and data stores often utilize different ECC schemes due to optimized memory organizations.

5 Separate tag store requires separate control logic and associated data paths and power requirements.

Utilizing DRAM as cache memory typically does not exist because it is difficult to use efficiently as cache memory and DRAM is not well suited to be reconfigured as scratch memory or fast system memory due to complexities in a controller that supports a separate tag store.

10 Additionally, tag store is often highly specialized memory that provides no value if the data store is reconfigured away from cache usage.

Techniques described herein provide the ability to utilize a new memory hierarchy transparently as cache memory and allows the new memory hierarchy to be reconfigured as scratch/fast memory as workloads and/or operating systems support fast/scratch memory.

15 Described herein are techniques to utilize 72-byte Error Correcting Code (ECC) support without requiring changes for cache usage. This may result in a simplified memory controller architecture while supporting both cache and non-cache usages.

More specifically, the techniques described herein support implementation of a cache memory that may utilize standard dynamic random access memory (DRAM) modules and stores  
20 data and tags in the same memory page. Traditional caches use special tag storage that stores tags separately and does not allow tag or data storage to be reconfigured as scratch/fast system memory. A new and more efficient ECC code may be used for cache data in order to free bits (that had been used as ECC bits) for use as tag storage.

In one embodiment, a new data arrangement within the cache data memory page may be  
25 used to aggregate the tag bits together for storage. In one embodiment, specific cache organizations and transaction flows are utilized to efficiently fetch tags of a cache set at one time and provide optimized accesses for subsequent cache data accesses.

External memories typically support Single x4 Error Correction-Double x4 Error Detection (S4EC/D4ED) or "Chipkill." This protects a data word that is 144 bits wide and stored across a  
30 "rank" that includes 36 x4-bit devices if the 144 bits are 128 data bits and 16 check bits. In a monolithic memory a single die provides the data word. In these cases a chip fail scheme is not necessary, but Single Error Correction-Double Error Detection (SEC/DED) ECC may still be provided.

In one embodiment, a 72-bit data bus may be used with 8-bit bursts to transfer a 64-byte  
35 cache line. In one embodiment, wire SEC/DED may be provided. That is any single wire fault

of the 72-bit bus is correctable and any second wire fault is detectable. A single wire fault can result in three cases of bit errors for even/odd bit pairs that share a common wire. In one embodiment, a 144-bit may be striped into a 72-bit data bus with the ECC codes as illustrated in Figure 1.

5 **Figure 1** is a conceptual illustration of one embodiment of a data arrangement that may be utilized for transmitting cache lines of data to and from a physical memory device. In the example of Figure 1, if a wire has a failure (e.g., a stuck-at fault), any of the following may occur: 1) no error with both even/odd data values match the failed value; 2) single bit error when either even or odd data values match the failed value; 3) double bit error when neither even/odd  
 10 data values match the failed value. All combinations are correctable.

A second wire fault will generate similar cases and all combinations of the original wire bit errors with the second wire bit error cases are detectable without aliasing. In one embodiment, the SEC-DED code is constrained to a modified Hamming code using 12 bits to allow for four extra bits to be available for data in each data word compared to the traditional chip-kill codes.

15 In one embodiment, a wire error code is a bit error code or the XOR of the two bit error code pairs for a given wire. Wire error codes are unique for a wire fault to be correctable. Also, the XOR of any two-wire error codes from distinct wires cannot be another wire error code. Otherwise, some double wire faults would not be detected.

20 With these constraints, a sufficient ECC code can be constructed. Each 12-bit bit error code can be considered a base-4 number, and the 12-bit space can be partitioned by non-zero leading digits.

P1	P2	P3	
1XXXX X	2XXXX X	3XXXX X	1,024 in each of P1, P2 and P3 with 1k <sup>2</sup> pair possibilities.
01XXX X	02XXX X	03XXX X	256 in each of P1, P2 and P3, 64k pair possibilities.
001XX X	002XX X	003XXX	32 in each of P1, P2 and P3
0001XX	0002XX	0003XX	16 in each of P1, P2 and P3
00001X	00002X	00003X	
000001	000002	000003	
000000	000000	000000	

Table 1

To generate wire error codes for a given wire, a value is chosen from P1 and P2 in the same row for the bit error codes, and the values are XORed, which will be in the P3 space for that row. This gives the final wire error code. A search program may be utilized to search for bit error code pairs such that the resulting wire error codes satisfy the SEC-DED constraints discussed above. In one embodiment 72 pairs are utilized to provide the desired level of ECC protection.

Referring back to Figure 1, data bits 180 can provide 128 bits of data over wire 110 using even bits 120 and odd bits 140. Similarly, ECC protection may be provided with 12 check bits 160 also over wire 110 using even bits 120 and odd bits 140. This allows additional data bits 170 to provide 4 bits of data over wire 110 using even bits 120 and odd bits 140. In one embodiment, data bits 170 may be used for cache tag information.

**Figure 2** is a conceptual diagram of one memory organization that may be utilized with the data configuration of Figure 1. Figure 2 illustrates a 256-byte memory page that may hold four 64-byte lines and associated 12-byte ECC bits per 16 bits of data. Using a traditional organization and the data configuration of Figure 1, there would be four unused bits per 16-byte data word distributed evenly with the data words. Other memory page sizes, line sizes, data sizes, etc can be supported.

In the example of Figure 2, memory page 200 includes four cache lines (e.g., 230, 240, 250 and 260) as well as ECC bits 280 and cache tag bits 220. The same data bits can be utilized in different organizations.

For a caching organization, it would be desirable to have all of the tag bits together to provide more efficient fetching of tags required for a lookup. Using the 12-byte ECC approach, there will be 16 spare bits per 64-byte line, providing 64 bits for tag storage for the four cacheline memory page.

The data organization of Figure 2 provides an aggregation of 32 bits to accommodate tags in the first data word of the memory page. The remaining bits of the memory page are assigned to data and ECC such that hardware requirements are simplified for unpacking of data and ECC when accessing any of the four 64-byte lines.

With the organization of Figure 2, the tags can be fetched in on short access and the remaining data words of any 64-byte line are accessible in at most four additional data words. In this organization, a minimum set of byte lanes requires simple hardware to accommodate tag or 64-byte line data. This organization is usable for non-cache uses with only small penalties that occur when accessing one of four 64-byte lines. In the example of Figure 2, unused spares may be utilized in a way to minimize hardware requirements. The concept of aggregating spare bits to the front of the page may be extended for variations in tag size and line size.

In one embodiment, the DRAM memory that is used as a memory cache is outside of a processor coherency domain and may support memory consistency and may use the MEI model. Using the organization of Figure 2, tag lookups and updates are in the same page as well as cached data fills, hits and evict accesses. Utilizing a page open policy for near memory scheduling allows relatively low power and relatively

**Figure 3** is a block diagram of one embodiment of an electronic system. The electronic system illustrated in Figure 3 is intended to represent a range of electronic systems (either wired or wireless) including, for example, desktop computer systems, laptop computer systems, cellular telephones, personal digital assistants (PDAs) including cellular-enabled PDAs, set top boxes, tablet devices, etc. Alternative electronic systems may include more, fewer and/or different components.

Electronic system 300 includes bus 305 or other communication device to communicate information, and processor 310 coupled to bus 305 that may process information. While electronic system 300 is illustrated with a single processor, electronic system 300 may include multiple processors and/or co-processors. Electronic system 300 further may include random access memory (RAM) or other dynamic storage device 320 (referred to as main memory), coupled to bus 305 and may store information and instructions that may be executed by processor 310. Main memory 320 may also be used to store temporary variables or other intermediate information during execution of instructions by processor 310.

In one embodiment, electronic system 300 includes DRAM cache memory 325 coupled with one or more of processor(s) 310. DRAM cache memory 325 may be configured to operate in the manner described above to provide cache memory to one or more of processor(s) 310 with a memory device that is external to the processor(s).

Electronic system 300 may also include read only memory (ROM) and/or other static storage device 330 coupled to bus 305 that may store static information and instructions for processor 310. Data storage device 340 may be coupled to bus 305 to store information and instructions. Data storage device 340 such as a magnetic disk or optical disc and corresponding drive may be coupled to electronic system 300.

Electronic system 300 may also be coupled via bus 305 to display device 350, such as a cathode ray tube (CRT) or liquid crystal display (LCD), to display information to a user. Alphanumeric input device 360, including alphanumeric and other keys, may be coupled to bus 305 to communicate information and command selections to processor 310. Another type of user input device is cursor control 370, such as a mouse, a trackball, or cursor direction keys to communicate direction information and command selections to processor 310 and to control cursor movement on display 350.

Electronic system 300 further may include network interface(s) 380 to provide access to a network, such as a local area network. Network interface(s) 380 may include, for example, a wireless network interface having antenna 385, which may represent one or more antenna(e). Network interface(s) 380 may also include, for example, a wired network interface to  
5 communicate with remote devices via network cable 387, which may be, for example, an Ethernet cable, a coaxial cable, a fiber optic cable, a serial cable, or a parallel cable.

In one embodiment, network interface(s) 380 may provide access to a local area network, for example, by conforming to IEEE 802.11b and/or IEEE 802.11g standards, and/or the wireless network interface may provide access to a personal area network, for example, by conforming to  
10 Bluetooth standards. Other wireless network interfaces and/or protocols can also be supported.

IEEE 802.11b corresponds to IEEE Std. 802.11b-1999 entitled "Local and Metropolitan Area Networks, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band," approved  
15 September 16, 1999 as well as related documents. IEEE 802.11g corresponds to IEEE Std. 802.11g-2003 entitled "Local and Metropolitan Area Networks, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 4: Further Higher Rate Extension in the 2.4 GHz Band," approved June 27, 2003 as well as related documents. Bluetooth protocols are described in "Specification of the Bluetooth System: Core, Version 1.1,"  
20 published February 22, 2001 by the Bluetooth Special Interest Group, Inc. Associated as well as previous or subsequent versions of the Bluetooth standard may also be supported.

In addition to, or instead of, communication via wireless LAN standards, network interface(s) 380 may provide wireless communications using, for example, Time Division, Multiple Access (TDMA) protocols, Global System for Mobile Communications (GSM) protocols, Code Division, Multiple Access (CDMA) protocols, and/or any other type of wireless  
25 communications protocol.

Reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same  
30 embodiment.

While the invention has been described in terms of several embodiments, those skilled in the art will recognize that the invention is not limited to the embodiments described, but can be practiced with modification and alteration within the spirit and scope of the appended claims. The description is thus to be regarded as illustrative instead of limiting.

CLAIMS

What is claimed is:

1. An apparatus comprising:
  - a processor core;
  - 5 a bus coupled to the processor core;
  - a dynamic random access memory (DRAM) coupled with the processor core via the bus, the DRAM to operate as a cache memory for the processor core, the processor core to transmit to the DRAM a block of data as even and odd pairs of bits, the block of data having N error correcting code (ECC) bits and  $11*N$  data bits.
- 10 2. The apparatus of claim 1 wherein N equals 12.
3. The apparatus of claim 2 wherein the processor core causes two or more cache lines to be stored in a memory page with tag bits aggregated together within the page.
- 15 4. The apparatus of claim 3 wherein the number of cache lines in the memory page is four.
5. The apparatus of claim 3, wherein each cache line comprises a block of data, the page of memory comprising four tag bytes at a beginning of the page, and ECC bits for each cache line stored as a portion of every eighth byte of the page.
- 20 6. The apparatus of claim 1, wherein ECC codes are constructed via a partitioned code space.
- 25 7. The apparatus of claim 6, wherein the partitioned code space is partitioned by leading non-zero digit values.
8. The apparatus of claim 7 wherein the number base is 4.
- 30 9. A method comprising:
  - operating a dynamic random access memory (DRAM) as a cache memory coupled with a processor core;

transmitting to the DRAM a block of data as even and odd pairs of bits with the processor core, the block of data having N error correcting code (ECC) bits and  $11*N$  data bits; and causing two or more cache lines to be stored in a memory page with tag bits aggregated together within the page.

5

10. The method of claim 9 wherein N equals 12.

11. The method of claim 9 wherein the number of cache lines in the memory page is four.

10

12. The method of claim 11, wherein each cache line comprises a block of data, the page of memory comprising four tag bytes at a beginning of the page, and ECC bits for each cache line stored as a portion of every eighth byte of the page.

15

13. The method of claim 9, wherein ECC codes are constructed via a partitioned code space.

14. The method of claim 13, wherein the partitioned code space is partitioned by leading non-zero digit values.

20

15. The method of claim 14 wherein the number base is 4.

16. An article of manufacture comprising a non-transitory computer-readable medium having stored thereon instructions that, when executed, cause one or more processors to:

25

operate a dynamic random access memory (DRAM) as a cache memory coupled with a processor core;

transmit to the DRAM a block of data as even and odd pairs of bits with the processor core, the block of data having N error correcting code (ECC) bits and  $11*N$  data bits; and

30

cause two or more cache lines to be stored in a memory page with tag bits aggregated together within the page.

17. The article of claim 16 wherein N equals 12.

18. The article of claim 16 wherein the number of cache lines in the memory page is four.

35

19. The article of claim 18, wherein each cache line comprises a block of data, the page of memory comprising four tag bytes at a beginning of the page, and ECC bits for each cache line stored as a portion of every eighth byte of the page.

5

20. The article of claim 16, wherein ECC codes are constructed via a partitioned code space.

21. The article of claim 20, wherein the partitioned code space is partitioned by leading non-zero digit values.

10

22. The article of claim 21 wherein the number base is 4.

23. A system comprising:

15

a transceiver having an antenna;

a processor core coupled with the transceiver;

a bus coupled to the processor core;

a dynamic random access memory (DRAM) coupled with the processor core via the bus, the DRAM to operate as a cache memory for the processor core, the processor core to transmit to the DRAM a block of data as even and odd pairs of bits, the block of data having N error correcting code (ECC) bits and  $11*N$  data bits.

20

24. The system of claim 23 wherein N equals 12.

25

25. The system of claim 24 wherein the processor core causes two or more cache lines to be stored in a memory page with tag bits aggregated together within the page.

26. The system of claim 25 wherein the number of cache lines in the memory page is four.

30

27. The system of claim 25, wherein each cache line comprises a block of data, the page of memory comprising four tag bytes at a beginning of the page, and ECC bits for each cache line stored as a portion of every eighth byte of the page.

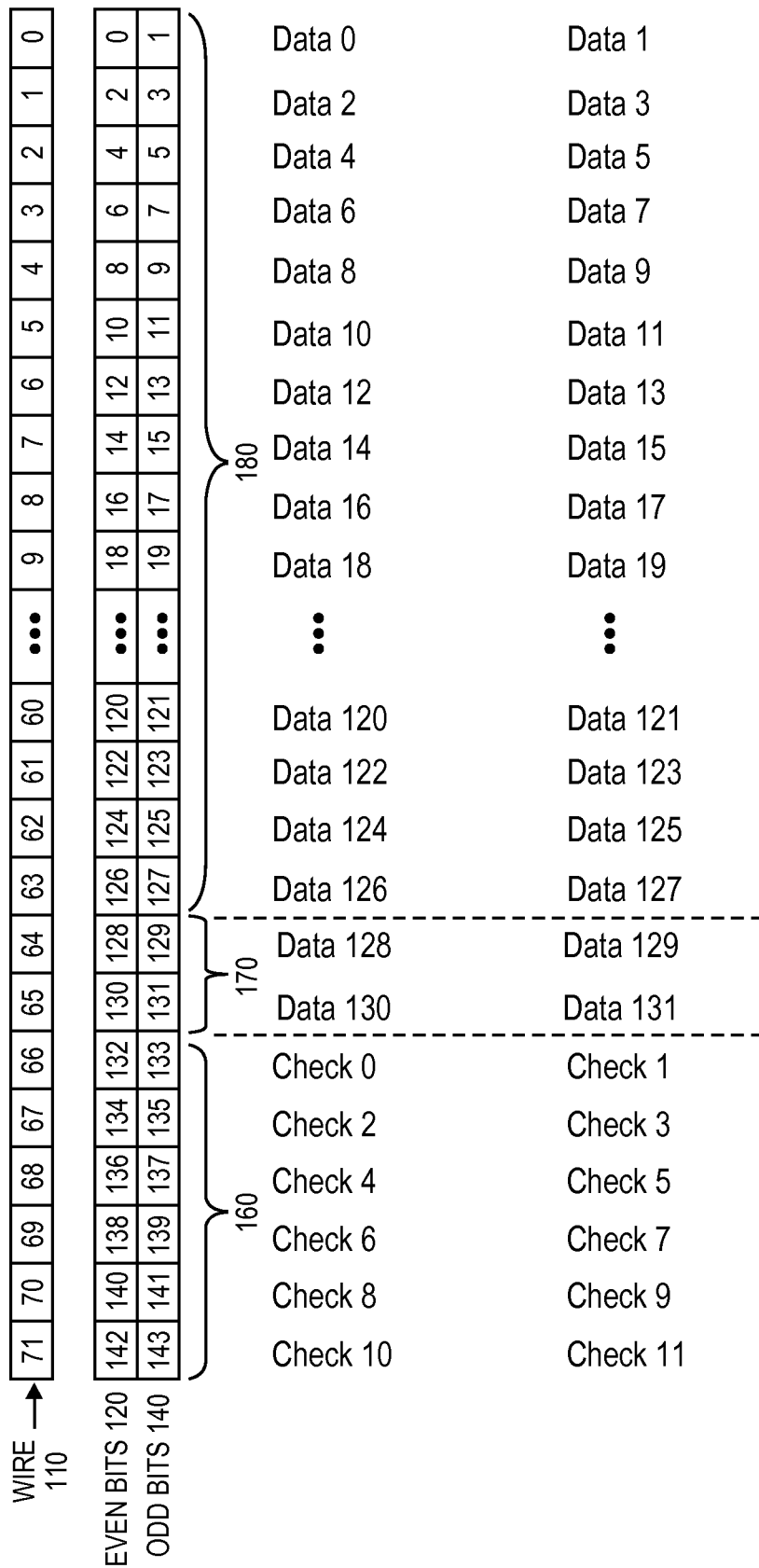


FIG. 1

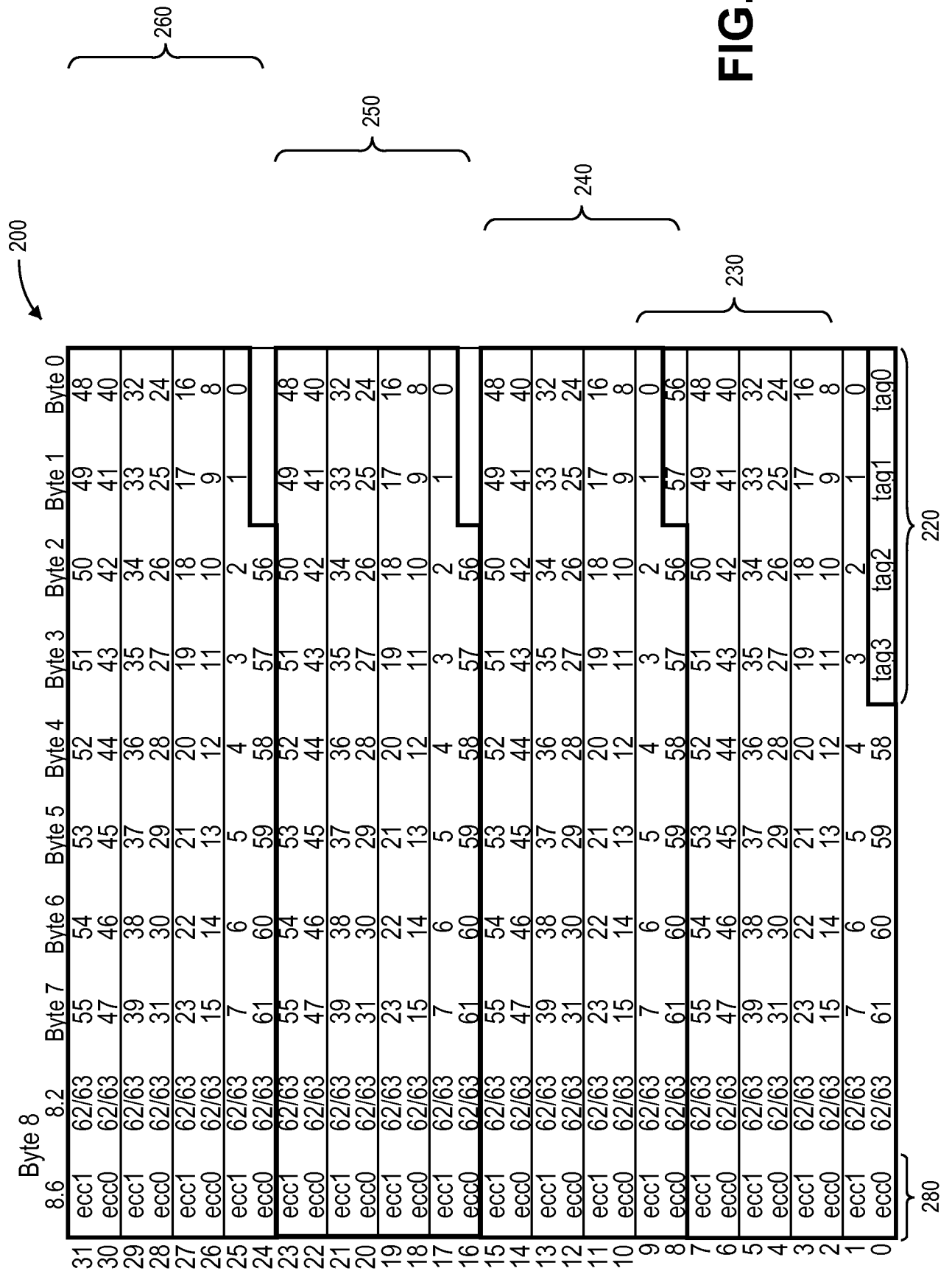
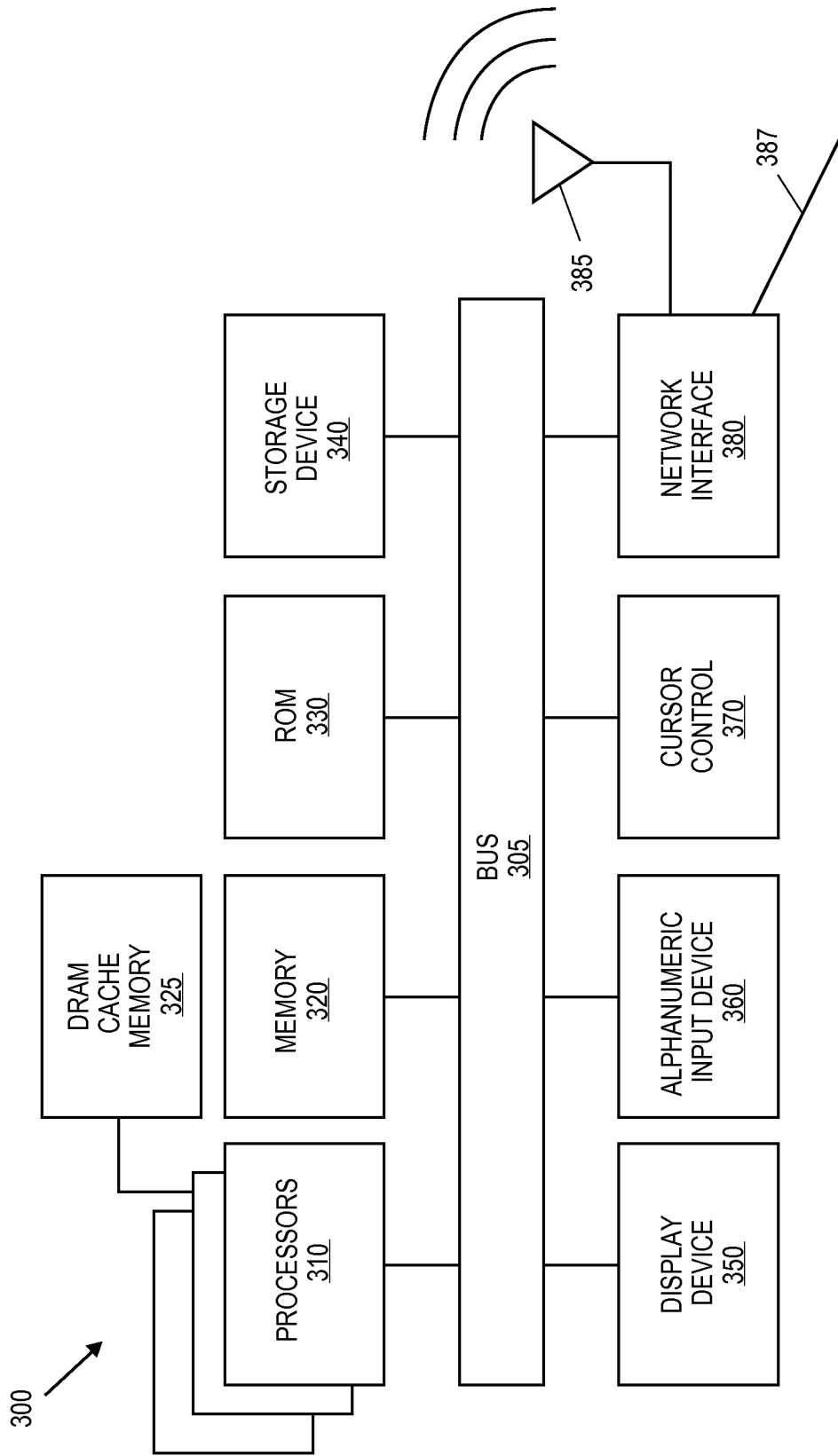


FIG. 2



**FIG. 3**