Why ECC & Predictive Maintenance Make a Better LPDDR4 DRAM

DRAM is critical for many modern digital ecosystems, particularly to drive computational workloads. However, though DRAMs excel in fast data access, they grapple with data integrity challenges. Specifically, flipped bits cause data corruption that can occur as a result of electrical interference, temperature changes, manufacturing defects, or even cosmic rays.

Although these incidents are infrequent, their implications can be far-reaching, especially in mission-critical and large-scale applications as seen in **Figure 1**.

Figure 1. Satellites and other space technology are vulnerable to flipped bits. Image courtesy of Pixabay.

Data corruption can degrade performance, compromise reliability, and, in worst-case scenarios, lead to system failure. The only practical way to avoid single-bit errors is to use error correction algorithms such as error correction code (ECC).

This white paper will review single-bit errors and how they impact the reliability of DRAM, followed by a discussion of embedded ECC. In addition, the paper will explore the benefits and applications of embedded ECC and introduce the Intelligent Memory LPDDR4 (along with frequently asked questions).

Impact of Single-Bit Errors and DRAM Reliability

According to a field study by the University of Toronto, 25,000 to 70,000 ECC-correctable single-bit errors occur per megabit of DRAM per 1 billion hours of operation.

The effects of single-bit errors are typically transient and take the form of difficult-to-repeat single-data bit flips, in which the value of a bit spontaneously flips from one state to the other. When bit errors go undetected or uncorrected, they can lead to data corruption, which in turn can cause flawed outcomes, system crashes, or complete failure.

Although not all DRAM single-bit errors cause a system crash, the application software may become unstable, or critical data can be unpredictably altered. In addition, the wrong data can pass through to external media, resulting in unrecoverable data errors. This random effect shows up in different ways at unknown times.

Single-bit errors often appear under heavy stress or long-term use of the DRAM; after a reset, the systems work again until a new occurrence. Other sources for single-bit errors include temperature and voltage fluctuations that can cause memory cells to become less stable, and software errors such as bugs in the operating system or device drivers.

To mitigate single-bit errors and their impact on data corruption, you can take one of several approaches. For example, DRAMs are factory-tested by long burn-in-testing, functionality, and speed testing with different patterns and voltage variations.

Another approach involves ECC functionality integrated into DRAM products. Unlike simple parity-based error checking, which only detects errors, ECC can detect and correct single-bit errors, providing an additional layer of assurance for data integrity. ECC-embedded DRAMs have been adopted across numerous sectors requiring high data reliability, including servers and data centers, scientific computing, space, avionics, automotive, and industrial systems.

Error Correction Code

In ECC DRAM, an 8-bit ECC is typically added to every 64-bit data word, making a 72-bit-wide buffer. When writing data, the ECC logic calculates and stores the ECC value in an extra 8-bit memory space. When reading, the logic recalculates and compares the ECC value to the stored value. When identifying a discrepancy, the logic pinpoints and corrects the erroneous bit.

The ability to correct errors ensures the accuracy of the data retrieved from DRAM. Data integrity is enhanced, boosting system reliability by preventing system crashes or failures because of memory errors.

ECC-Embedded DRAM

A standard DRAM unit contains eight memory chips, but in an ECC-embedded DRAM, an additional chip performs the error correction. The result of embedding the ECC chip with the eight memory chips is faster recognition and correction of flipped bits.

Benefits of ECC-Embedded DRAM

There are four specific benefits of using ECC-embedded DRAM:
- Reliable performance in large-scale and critical computing applications
- Cost-effective mitigation of data corruption
- Extended DRAM lifespan
- Support for preventive maintenance

Applications of ECC-Embedded DRAM

ECC-embedded DRAM is well suited to applications requiring high reliability, including situations in which data corruption would be catastrophic (for example, the data is in a remote location (making it difficult to make repairs) or the data is in extreme operating environments. Such applications would include the following:
- Servers and data centers
- Scientific computing and research
- Space and avionics systems
- Automotive systems
- Industrial systems

Servers and Data Centers

Servers and data centers house vast amounts of information and execute critical tasks every second — they are at the heart of the digital economy. Showin in **Figure 2**, data corruption resulting from many sources, including electrical and magnetic interference, poses a major challenge for servers and data centers. As servers handle sensitive data and perform complex calculations, data corruption can lead to consequences, such as financial loss or damage to an organization's reputation.

Figure 2. *Flipped bits can be disastrous for data centers.*
Image provided courtesy of [*Pixabay*](Pixabay)*.*

ECC DRAM in servers and data centers ensures data integrity and system stability by detecting and correcting bit errors, making it invaluable for improving server reliability. As data centers grow and handle larger volumes of data in the evolving technological landscape, the importance of ECC DRAM becomes more pronounced. With cloud computing, big data, and AI processing becoming more prevalent, data corruption can have far-reaching effects on service availability and the validity of processing results. ECC DRAM provides a robust foundation for these advanced computing services.

Scientific Computing and Research

Scientific computing and research involves complex calculations and simulations that demand computational accuracy in fields such as physics, chemistry, meteorology, biology, artificial intelligence, and even economics. However, the intense computing processes involved can lead to system memory errors.

ECC DRAM's ability to detect and correct errors can be critical. In weather forecasting models, for instance, minor data errors can lead to drastically different predictions. Similarly, in computational biology, data errors can lead to inaccurate protein folding simulations with severe implications for drug discovery.

The use of ECC DRAM in scientific computing and research plays an essential role in maintaining the accuracy of these high-stakes computations.

Space and Avionics Systems

ECC DRAM can be crucial for space and avionics systems. In these systems, the hardware is exposed to high radiation levels, which can cause frequent memory errors. These systems also often operate in isolated environments where maintenance or immediate troubleshooting is impossible, making ECC DRAM's self-correcting ability invaluable.

Figure 3. Aircraft are an example of systems in which accurate memory is critical.
Image provided courtesy of Pixabay.

Cosmic rays, in particular, can cause single-event upsets (SEUs) in memory systems. SEUs are a change of state caused by ions or electromagnetic radiation striking a sensitive node in a microelectronic device, causing data corruption. By incorporating ECC DRAM in space and avionics systems (as shown in **Figure 3**), these SEUs can be detected and corrected automatically, ensuring continued operation and mitigating the risk of mission failure.

Automotive Systems

The automotive and industrial sectors also heavily utilize ECC DRAM, demanding robust fault tolerance and reliability due to the critical nature of the applications and their harsh operating environments. In an automotive context, advanced driver-assistance systems (ADAS) and infotainment systems can benefit from ECC DRAM's increased reliability and data integrity.

Industrial Systems

In the industrial sector, systems controlling machinery, logistics, and process automation must function reliably under various conditions, e.g., temperature fluctuations, dust, and vibration. ECC-enabled DRAM ensures that these systems can handle potential memory errors caused by challenging environments, maintaining system stability, and preventing possible damage or accidents. The need for reliable memory systems will only grow as the automotive and industrial sectors increase their use of automation.

Intelligent Memory's LPDDR4

The Intelligent Memory (IM) LPDDR4 with integrated ECC DRAM, shown in **Figure 4**, is a Low Power Double Data Rate DRAM integrating error-correction capabilities that consume less energy than its predecessors, engineered to operate over the industrial and high-temperature ranges. It is entirely Joint Electron Device Engineering Council (JDEC) compliant and provides a mode register (MR) setting that allows end users to turn the ECC function ON/OFF as needed.
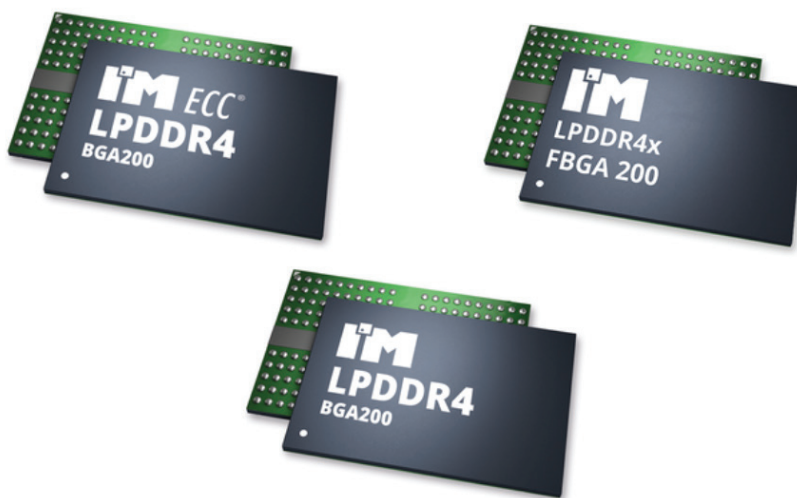

Figure 4. The IM LPDDR4 DRAM with embedded ECC.

Below are key features of IM's LPDDR4 DRAM:
- Available in densities from 4 Gb to 64 Gb
- Embedded ECC error correction for LPDDR4 4Gb x 32 and 8Gb x 32
- Fast data rates up to 4.266 Gbps
- Low Voltage Power Supply 1.8 V and I/O at 1.1 V for the LPDDR4
- Low I/O voltage of 0.6 V for the LPDDR4x
- ZQ calibration
- Fully synchronous operation
- Long-term support

The ECC-Embedded IM LPDDR4 and Preventive Maintenance

In the context of DRAM, preventive maintenance means to take measures to prevent system failure due to flipped bits before data corruption has occurred.

Preventive maintenance is a powerful tool for achieving reliability, whether it's applied to rotating equipment in a manufacturing facility or DRAM memory that holds critical data for an aircraft carrying hundreds of passengers. In fact, Plant Engineering reports that, as of 2021, 88% of industrial facilities follow a preventive maintenance strategy.

The consequences of data corruption run the spectrum, ranging from loss of productivity to loss of reputation, lawsuits, and even corporate disaster. The IM LPDDR4 ECC DRAM records ECC events and provides failure data as an integral part of preventive maintenance. The data from the ECC event counter can be analyzed for the mean time between failure (MTBF) and point to a system that is rapidly becoming unstable; steps can be taken to maintain or replace the memory *before* failure occurs. In short, wise use of error data prevents failure, and prevention is always better than remedy.

How Embedded ECC Works in the LPDDR4

IM's LPDDR4 DRAM implements an on-chip ECC circuit per channel. The ECC chip uses a 64-bit single error correction, double error detection (SEC-DED) code that detects and corrects all single-bit errors and detects double-bit errors, including those introduced by SER events such as cosmic rays and alpha particles. To maximize reliability, the ECC is implemented across 64-bit data quantum using 8 ECC parity bits for 72 bits per ECC quantum.

Intelligent Memory Solution: Focusing on Preventive Maintenance

ECC-embedded DRAM both recognizes and repairs single-bit errors, and the data collected about flipped bits supports preventive maintenance. The data can be used to set thresholds that, when exceeded, can alert applicable personnel when a memory system is becoming unstable and needs either repair or replacement.

The IM LPDDR4 with embedded ECC is an ideal solution for mission-critical applications in which corrupt data can be disastrous. Fully JEDEC compliant, Intelligent Memory's LPDDR4 embeds error-correction capabilities that consume less energy and perform even in rugged environments. Contact Intelligent Memory today to learn more about the IM LPDDR4.

Frequently Asked Questions

How does embedded ECC differ from conventional ECC solutions?

ECC error correction is commonly used in high-end industrial applications and servers. It requires an ECC-capable memory controller with an extra-wide data bus width, e.g., 72 bits (64 data bits + 8 check bits). The memory controller generates the required additional check-bits for the data and writes the extra-wide data word to the memory. Upon a read command, the memory controller verifies the data word integrity, checks bits, and performs the correction algorithm. The typical method of ECC correction requires multiple DRAMs to be accessed in parallel to achieve an extra-wide bit width. On server-memory modules, for example, up to 18 DRAM components with 4 data lines each are placed in parallel to achieve a 72-bit data bus.

With IM's ECC DRAM, check-bit generation, verification, and correction are all performed inside the same memory device. Each single ECC DRAM performs the error correction. Thus, it does not require ECC-capable processors, controllers, or an extra-wide data bus between the controller and the DRAM. Because the ECC DRAM components are fully JEDEC-compliant, they can serve as drop-in replacements for conventional DRAM memory. Any existing application built with conventional DRAM can be equipped with error-correction functionality.

How do I know whether the error correction (ECC) is really working inside the LPDDR4 ECC DRAM chip?

The verification takes the proper steps: setting the ECC function, forcing the DRAM cell failed, and then reading the ECC event counter. The following steps explain how to do that.

1. The ECC function type can be set by MR#33 (Refer to **Table 1**). ECCON (OP7R/W) can set ECC function ON when OP[7] = 1. For ERRON (OP6R/W) set OP[6] = 0. (Note that the OP [6:7] = "01" means we want to count 1-bit ECC Event.)

2. For the DRAM cell failed, keep in mind that the DRAM chip is sound when it is shipped from the factory. It needs to generate a harsh environmental condition to force it to fail, such as using hot air to heat the DRAM chip when the system is in working status (the DRAM working temperature can be up to 95°C). Heating the DRAM to 105°C to 120°C can force DRAM cell fail because the higher temperature reduces the retention time of the DRAM cell.

3. Read the ECC event counter. How many 1-bit ECC events occurred can be checked by reading MR34 (Refer to **Table 2**)? The 8 bits can count a maximum of 255 ECC events.

MR#	Access	OP4	OP3	OP2	OP5W	OP1R	OP0R	OP6R/W	OP7R/W
33		Reserved			CLRECC	ERR TYPE		ERRON	ECCON

Table 1. MR33 (ER0)

MR#	Access	OP7	OP6	OP5	OP4	OP3	OP2	OP1	OP0
34	R	ECC Event Counter							

Table 2. MR34 (ER1)

Error correction code mentions "1-bit correction 2-bit detection." Can the LPDDR4 ECC DRAM detect 2-bit failure event?

Yes, this LPDDR4 ECC DRAM can be set to record 1-bit correction event or 2-bit error event. Please refer to the setting below.

To set the 2-bit detection ECC mode, use the MR#33 (see Table 1). Set ECCON (OP7R/W) to OP[7] = 1. The ECC function is still ON, and 1-bit errors still will be corrected. Next, set ERRON (OP6R/W) OP[6] = 1. The OP [6:7] = "11" means we want to count 2-bit error events.

Please, note that 1-bit ECC events are no longer counted in this mode. Instead, the ECC event counter will record how many 2-bit error events happened. It can be checked by reading MR34.

Besides reading the Mode Register (MR#34), is there any other way to know if an ECC event happened?

Yes, some reserve pins are used to achieve this goal. The ERR_A,B signal indicates ECC event occurrence per each channel, and is the same interface with DQ (1.1 V LVSTL). Referring to **Figure 5**, Pin "A11" is ERR_A and "AB11" is ERR_B. Each pin Indicates ECC event occurrence per each channel. Signal "LOW" means no 1-bit error, while signal "HIGH" means 1-bit Error Detected and Corrected or 2-bit Error Detected.

Pin Configurations

200-ball x32 Discrete Package, 0.80mm x 0.65mm using MO-311

	1	2	3	4	5	6	7	8	9	10	11	12
A	NC	NC	V_{SS}	V_{DD2}	ZQ0			NC	V_{DD2}	V_{SS}	ERR_A	NC
B	NC	DQ0_A	V_{DDQ}	DQ7_A	V_{DDQ}			V_{DDQ}	DQ15_A	V_{DDQ}	DQ8_A	NC
C	V_{SS}	DQ1_A	DMI0_A	DQ6_A	V_{SS}			V_{SS}	DQ14_A	DMI1_A	DQ9_A	V_{SS}
D	V_{DDQ}	V_{SS}	DQS0_T_A	V_{SS}	V_{DDQ}			V_{DDQ}	V_{SS}	DQS1_T_A	V_{SS}	V_{DDQ}
E	V_{SS}	DQ2_A	DQS0_C_A	DQ5_A	V_{SS}			V_{SS}	DQ13_A	DQS1_C_A	DQ10_A	V_{SS}
F	V_{DD1}	DQ3_A	V_{DDQ}	DQ4_A	V_{DD2}			V_{DD2}	DQ12_A	V_{DDQ}	DQ11_A	V_{DD1}
G	V_{SS}	ODT_CA_A	V_{SS}	V_{DD1}	V_{SS}			V_{SS}	V_{DD1}	V_{SS}	NC	V_{SS}
H	V_{DD2}	CA0_A	NC	CS0_A	V_{DD2}			V_{DD2}	CA2_A	CA3_A	CA4_A	V_{DD2}
J	V_{SS}	CA1_A	V_{SS}	CKE0_A	NC			CK_t_A	CK_c_A	V_{SS}	CA5_A	V_{SS}
K	V_{DD2}	V_{SS}	V_{DD2}	V_{SS}	NC			NC	V_{SS}	V_{DD2}	V_{SS}	V_{DD2}
L												
M												
N	V_{DD2}	V_{SS}	V_{DD2}	V_{SS}	NC			NC	V_{SS}	V_{DD2}	V_{SS}	V_{DD2}
P	V_{SS}	CA1_B	V_{SS}	CKE0_B	NC			CK_t_B	CK_c_B	V_{SS}	CA5_B	V_{SS}
R	V_{DD2}	CA0_B	NC	CS0_B	V_{DD2}			V_{DD2}	CA2_B	CA3_B	CA4_B	V_{DD2}
T	V_{SS}	ODT_CA_B	V_{SS}	V_{DD1}	V_{SS}			V_{SS}	V_{DD1}	V_{SS}	RESET_n	V_{SS}
U	V_{DD1}	DQ3_B	V_{DDQ}	DQ4_B	V_{DD2}			V_{DD2}	DQ12_B	V_{DDQ}	DQ11_B	V_{DD1}
V	V_{SS}	DQ2_B	DQS0_C_B	DQ5_B	V_{SS}			V_{SS}	DQ13_B	DQS1_C_B	DQ10_B	V_{SS}
W	V_{DDQ}	V_{SS}	DQS0_T_B	V_{SS}	V_{DDQ}			V_{DDQ}	V_{SS}	DQS1_T_B	V_{SS}	V_{DDQ}
Y	V_{SS}	DQ1_B	DMI0_B	DQ6_B	V_{SS}			V_{SS}	DQ14_B	DMI1_B	DQ9_B	V_{SS}
AA	NC	DQ0_B	V_{DDQ}	DQ7_B	V_{DDQ}			V_{DDQ}	DQ15_B	V_{DDQ}	DQ8_B	NC
AB	NC	NC	V_{SS}	V_{DD2}	V_{SS}			V_{SS}	V_{DD2}	V_{SS}	ERR_B	NC

Figure 5. *IM LPDDR4 BGA200 Pin Configurations.*

INTELLIGENT MEMORY