Technical Brief

Tech Brief Series: Analyzing Managed Flash Device[1] Lifetime Reliability

Part 4: Conducting a Lifetime Reliability Analysis

A robust, embedded managed flash device storage design requires a thorough lifetime data reliability analysis to ensure that user data in NAND flash memory maintains its integrity over a long lifetime. The focus of this tech brief, and the final installment of the "Analyzing Managed Flash Device Lifetime Reliability" series, presents a process for conducting an accurate lifetime reliability analysis. It provides key principles discussed in Parts 1, 2 and 3 of this series. Factors that may affect this analysis include the overall write endurance, which determines the length at which data can be retained in NAND flash memory, as well as the data retention value at the end of a device's life, which determines the refresh functionality method that should be used.

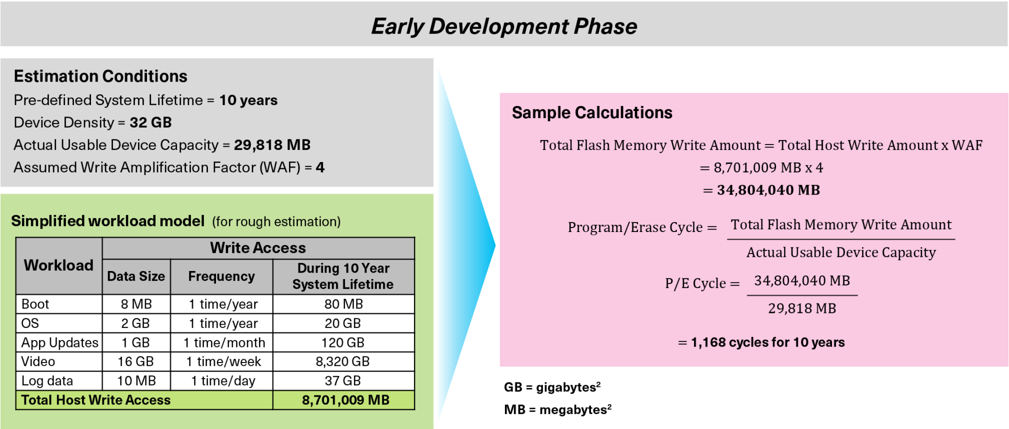
The steps covered in this tech brief provide user instruction for performing a proper NAND flash memory lifetime reliability analysis. It is important to note that certain data *needs to be* obtained from the managed flash supplier, as highlighted in this tech brief. It is also highly encouraged that users and NAND flash memory suppliers work together to review each lifetime reliability analysis for accuracy as small variations in certain parameters, due to miscalculations, can negatively affect the lifetime of NAND flash memory.

Step 1: Check and Calculate Total Lifetime Program/Erase Cycles

The first step to deliver an accurate NAND flash memory lifetime reliability analysis involves calculating and extrapolating the amount and frequency of write operations used for various system workloads at the managed flash device interface.

As a classic development cycle requires hardware development first, firmware driver development second, and application-level software development third, a managed flash device uses different approaches to estimate total lifetime writes that are dependent on these development phases. For example, in the early development phase, rough estimations can be made that could become the foundation for a lifetime reliability analysis. Estimations involving write workloads and the Write Amplification Factor (WAF) are more difficult.

The Figure 1 charts show the write endurance of a surveillance camera in an early development phase analysis where the left-side charts depict estimated conditions for simplified workload models, while the right-side chart provides sample calculations of the analysis. A pre-defined system lifetime is needed to be set and based on the life expectancy of the system to meet this calculation. As such, it is highly recommended that users consult with the managed flash device vendor for a suitable assumed WAF value.



Early Development Phase

Estimation Conditions
Pre-defined System Lifetime = **10 years**
Device Density = **32 GB**
Actual Usable Device Capacity = **29,818 MB**
Assumed Write Amplification Factor (WAF) = **4**

Simplified workload model (for rough estimation)

Workload	Write Access		
	Data Size	Frequency	During 10 Year System Lifetime
Boot	8 MB	1 time/year	80 MB
OS	2 GB	1 time/year	20 GB
App Updates	1 GB	1 time/month	120 GB
Video	16 GB	1 time/week	8,320 GB
Log data	10 MB	1 time/day	37 GB
Total Host Write Access			**8,701,009 MB**

Sample Calculations

$$\text{Total Flash Memory Write Amount} = \text{Total Host Write Amount x WAF}$$
$$= 8{,}701{,}009\text{ MB x }4$$
$$= \mathbf{34{,}804{,}040\ MB}$$

$$\text{Program/Erase Cycle} = \frac{\text{Total Flash Memory Write Amount}}{\text{Actual Usable Device Capacity}}$$

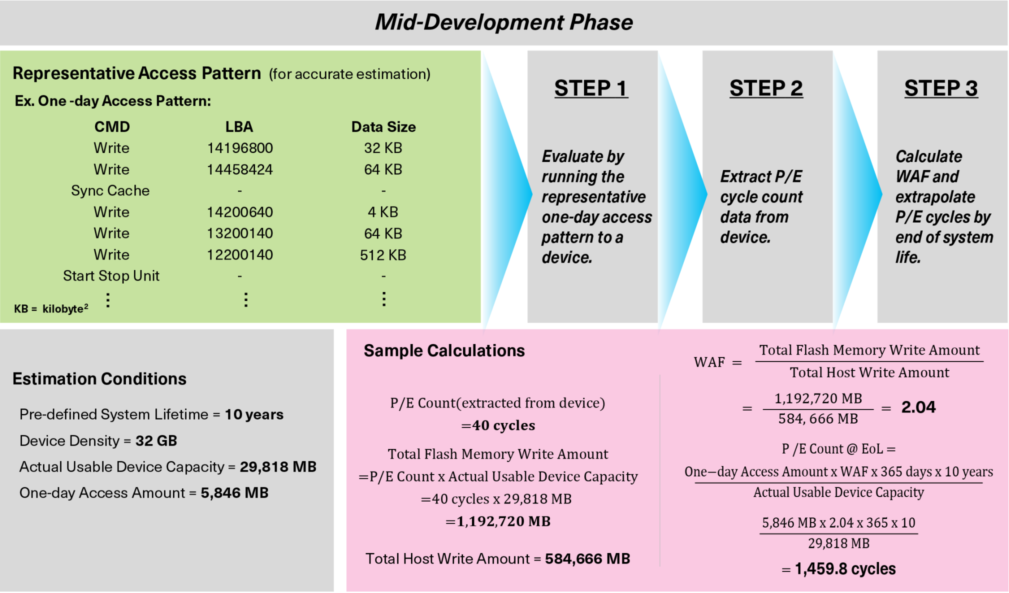
$$\text{P/E Cycle} = \frac{34{,}804{,}040\text{ MB}}{29{,}818\text{ MB}}$$

$$= \mathbf{1{,}168\ cycles\ for\ 10\ years}$$

GB = gigabytes[2]
MB = megabytes[2]

Figure 1. Example video surveillance use case in the early development phase of the Program/Erase analysis

A small variation in WAF (e.g., 1.5 versus 2) can significantly change the total flash memory lifetime write count, which in turn, can affect data retention time. A more accurate method for obtaining the total Program/Erase (P/E) cycles is to use the access pattern logs from managed flash workloads to calculate total writes (Figure 2). This method also yields a more accurate WAF as presented in Part 1 of this series. This type of reliability analysis is typically performed when application-level software is reasonably stable, such as the mid to late development stages. To obtain the needed logs for this analysis, contact the local KIOXIA representative for assistance. It should be noted that the actual usable device capacity information can be referenced from the e-MMC[3] or UFS[4] managed flash device datasheet.



Mid-Development Phase

Representative Access Pattern (for accurate estimation)

Ex. One -day Access Pattern:

CMD	LBA	Data Size
Write	14196800	32 KB
Write	14458424	64 KB
Sync Cache	-	-
Write	14200640	4 KB
Write	13200140	64 KB
Write	12200140	512 KB
Start Stop Unit	-	-
⋮	⋮	⋮

KB = kilobyte[2]

STEP 1

Evaluate by running the representative one-day access pattern to a device.

STEP 2

Extract P/E cycle count data from device.

STEP 3

Calculate WAF and extrapolate P/E cycles by end of system life.

Estimation Conditions

Pre-defined System Lifetime = **10 years**

Device Density = **32 GB**

Actual Usable Device Capacity = **29,818 MB**

One-day Access Amount = **5,846 MB**

Sample Calculations

P/E Count(extracted from device)
=**40 cycles**

Total Flash Memory Write Amount
=P/E Count x Actual Usable Device Capacity
=40 cycles x 29,818 MB
=**1,192,720 MB**

Total Host Write Amount = **584,666 MB**

$$\text{WAF} = \frac{\text{Total Flash Memory Write Amount}}{\text{Total Host Write Amount}} = \frac{1{,}192{,}720\ \text{MB}}{584{,}666\ \text{MB}} = \mathbf{2.04}$$

$$\text{P/E Count @ EoL} = \frac{\text{One-day Access Amount x WAF x 365 days x 10 years}}{\text{Actual Usable Device Capacity}}$$

$$\frac{5{,}846\ \text{MB x } 2.04 \text{ x } 365 \text{ x } 10}{29{,}818\ \text{MB}}$$

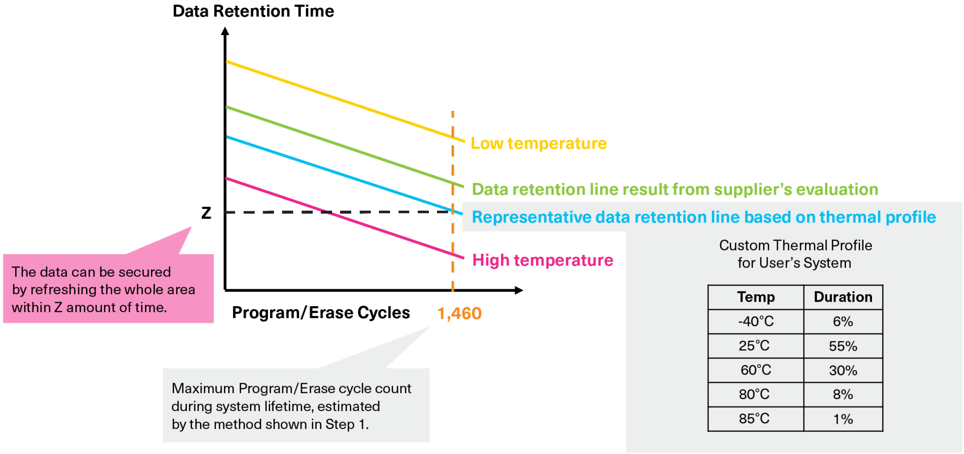
$$= \mathbf{1{,}459.8\ cycles}$$

Figure 2. Example use case in the mid to late development stages of the Program/Erase analysis

NOTE: *WAF results can fluctuate based on different types of workloads, such as random and sequential, and should be considered during the application-level software development design stage. Each workload WAF should be confirmed with the user's NAND flash memory vendor with the objective of achieving the lowest WAF possible for the best NAND flash memory lifetime.*

Step 2: Calculate Worse Case Data Retention

The next lifetime reliability calculation that needs to be determined is data retention, namely, the worst case scenario. However, before this part of the analysis can be calculated, the system's thermal profile must be obtained. As temperature increases, data retention decreases, as presented in Part 2 of this series. The thermal profile is user-customizable or users can simply request data for various constant temperatures, as shown by the blue and pink plots in Figure 3. Users can also obtain data from the NAND flash memory supplier's evaluation data as depicted by the green plot in Figure 3. For this calculation, it is highly recommended to use temperatures that pertain specifically to each managed flash device.



The data can be secured by refreshing the whole area within Z amount of time.

Maximum Program/Erase cycle count during system lifetime, estimated by the method shown in Step 1.

Custom Thermal Profile
for User's System

Temp	Duration
-40°C	6%
25°C	55%
60°C	30%
80°C	8%
85°C	1%

Figure 3. Example of a data retention analysis

NOTE: *Only one graph point is presented in Figure 3 where data retention is 1,460 cycles (equals Z). Requesting data retention for constant maximum P/E cycles at multiple temperatures is a common user request. Figure 3 is a generic reference to better show data retention versus P/E cycles versus temperature.*

At this stage of the lifetime reliability analysis, the worst case data retention, based on total P/E cycle count and system ambient temperature, needs to be calculated. Though Figure 3 represents data retention versus P/E cycles and ambient temperatures for surveillance cameras as a use case, the plots are theoretical values and not actual values. To obtain the actual values, users can acquire them directly from the managed flash device supplier.

In review of Figure 3, the worst case data retention has a total P/E cycle count of 1,460 after 10 years in conjunction with the custom system thermal profiles represented. These data points are used to obtain the worst case data retention values from the Y-axis (or value Z). As Figure 3 represents a theoretical plot example, users need to provide temperature and P/E cycle counts to the NAND flash memory supplier to get corresponding data retention values.

Step 3: Establish Proper Refresh Algorithm (if required)

At this stage of the lifetime reliability analysis, user now have the two main NAND flash memory reliability parameters that are needed to complete the analysis: (1) the total P/E count; and (2) the worst case data retention. These values can be compared to the managed flash device's specifications as a means to initially foresee any potential issues.

If the calculated data retention will not suffice the managed flash device design, refresh functionality can be used to prolong data retention, as presented in Part 3 of this series. Essentially, the entire memory space needs to be parsed to find the flash memory cells that need to be rewritten or refreshed. Using Figure 3 as a theoretical example, it is good practice to ensure that all memory space is parsed and refreshed before Z months expire, and highly recommended to work directly with the managed flash supplier to design an efficient refresh algorithm.

Summary

This concludes the "Analyzing Managed Flash Device Lifetime Reliability" series meant to be a tool for managed flash device users. The series enables users to gain a deeper understanding of the analysis that can be used to develop a robust lifetime data reliability study, as well as the steps required to conduct that study. KIOXIA engineers are ready to help guide users through the ins and outs of a managed flash device reliability analysis, and may help users to mitigate any unforeseen pitfalls that may occur. Contact the local KIOXIA representative.

General information for KIOXIA memory products is available here.

FOOTNOTES:

[1] A managed flash device combines raw NAND flash memory and an intelligent controller in one integrated package, enabling internal memory management.

[2] Definition of capacity - KIOXIA Corporation defines a kilobyte (KB) as 1,024 bytes, a megabyte (MB) as 1,000,000 bytes, a gigabyte (GB) as 1,000,000,000 bytes, a terabyte (TB) as 1,000,000,000,000 bytes and a petabyte (PB) as 1,000,000,000,000,000 bytes. A computer operating system, however, reports storage capacity using powers of 2 for the definition of 1Gbit = 2^{30} bits = 1,073,741,824 bits, 1GB = 2^{30} bytes = 1,073,741,824 bytes, 1TB = 2^{40} bytes = 1,099,511,627,776 bytes and 1PB = 2^{40} bytes = 1,125,899,906,842,624 bytes and therefore shows less storage capacity. Available storage capacity (including examples of various media files) will vary based on file size, formatting, settings, software and operating system, and/or pre-installed software applications, or media content. Actual formatted capacity may vary.

[3] Embedded MultiMediaCard (e-MMC) is a specification developed by JEDEC® for mobile applications. The current release is v5.1, published in February 2015.

[4] Universal Flash Storage (UFS) devices are based on the UFS specification, of which, the v4.0 specification is the current release issued by JEDEC and published in August 2022.

TRADEMARKS:

JEDEC is a registered trademark of the Joint Electron Device Engineering Council (JEDEC) Solid State Technology Association. All other company names, product names and service names may be trademarks of third-party companies.

DISCLAIMERS: