Everspin Technologies MRAM Optimizes System Energy Consumption

A technology such as MRAM can significantly reduce total system energy consumption compared to either EEPROM or flash. For many wireless and portable applications, especially in the growing Internet of Things, the energy budget (total power consumed over time) is a critical component. When calculating a design's power budget, engineers typically look at the rated power consumption of the device. However, other factors can come into play. For example, for non-volatile memory, the write current is very much higher than the read or standby current. Consequently, the write time needs to be considered in power-sensitive applications, especially in systems that sustain frequent memory writes. A technology such as MRAM, with fast-Write and power-up-to-Write times, can significantly reduce total system energy consumption compared to either EEPROM or flash. In this article, we compare system energy consumption in a typical data-acquisition system using flash, EEPROM or MRAM.

Overall, the comparisons showed that:

• Non-volatile memory write time is a major contributor to overall system energy consumption; thus, the shorter write time of MRAM can actually reduce total energy consumption.
• Further system energy reductions are possible using a power-gating architecture with MRAM, because its faster power-up to write time allows MRAM standby power to be reduced to zero.

Typical System

The schematic in *Figure 1* represents the low-voltage dropout regulator (LDO), microcontroller (MCU), non-volatile memory, and a decoupling capacitor typical in data-acquisition applications like medical monitors, data loggers, etc. Other system components, such as sensors and their power consumption, were not considered.
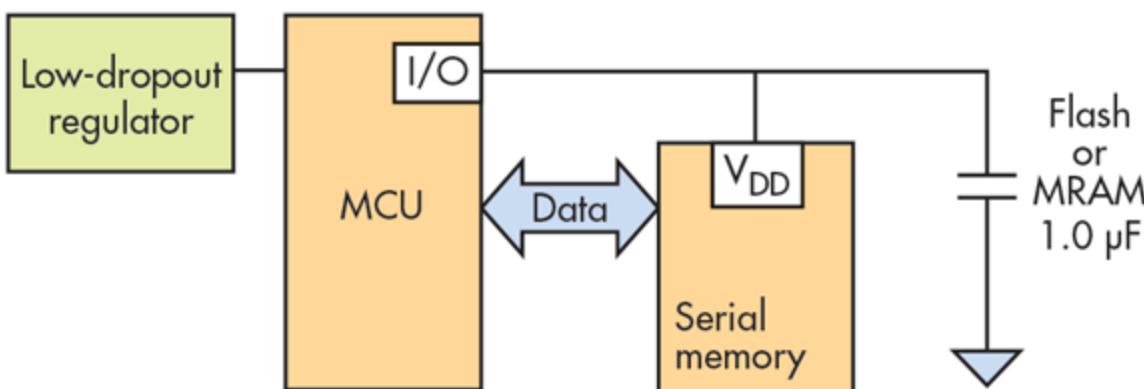
The MCU was assumed to be in a low power sleep state and with a periodic wakeup for data acquisition. The data acquired was stored in the non-volatile memory, and then the system returned to the sleep state.
We will compare non-volatile memory with an SPI interface and look only at Write operations, which typically consume much more power than Read operations. The number of data bytes that can be written is four less than the number of bytes on the SPI bus due to overhead of the Write command, the WREN bit, and two address bytes. The number of bytes written to the non-volatile memory was selected as 4 and 46. Four is perhaps most likely, representing the storage of one data-acquisition sample. Meanwhile, 46 is the optimum amount of data that can be written to an MRAM when powered from a 1.0-µF decoupling capacitor.

EST 2917   July, 2020

Power-Gating Considerations

A few quick calculations revealed that the decoupling capacitor is very important when power gating. The energy used to charge the capacitor from zero is significant. EEPROM can be powered directly from the I/O of a standard microcontroller, typically 4 mA. As a result, a small 0.1-µF capacitor was used for decoupling. MRAM and flash need more current than is available from a standard MCU I/O. Therefore, a larger decoupling capacitor is needed so that flash or MRAM could run on the energy stored in the device.

Phases of the Write Operation

The energy consumption of the non-volatile memory was calculated during the phases of the write operation *(Fig. 2)*:
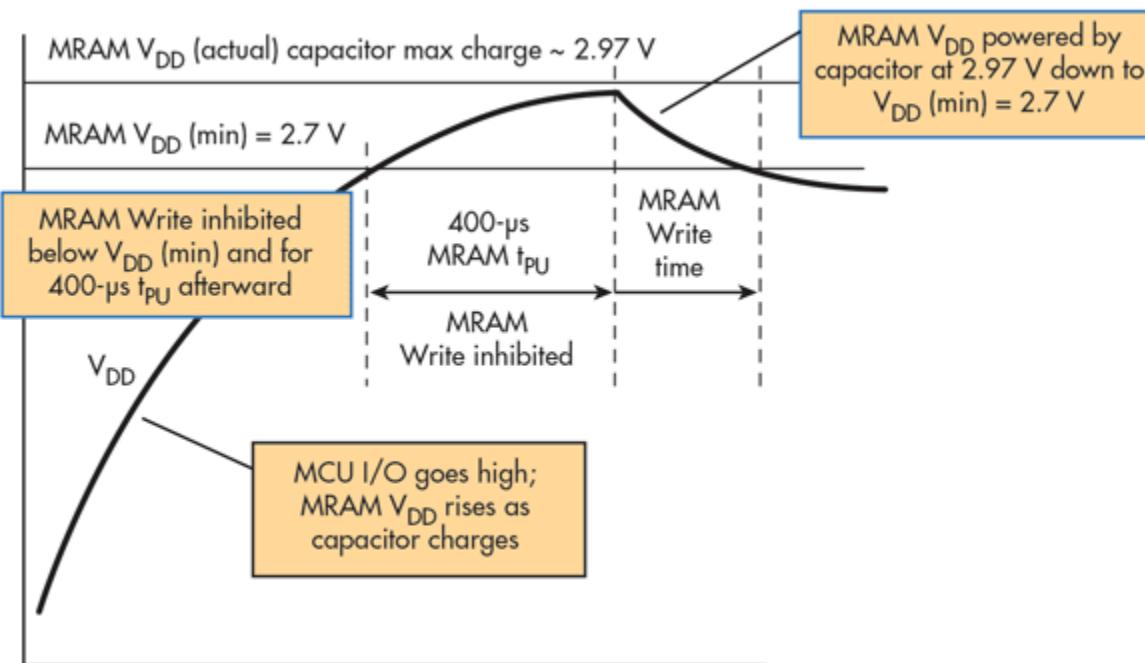
Rise time: During this phase, we assume that all of the energy goes into the decoupling capacitor and that the non-volatile memory consumes negligible energy.

Power-up time: Once the voltage on VDD is above a threshold, a small delay (tPU) is required for MRAM to become ready, but not for EEPROM or flash. During this phase, we assume that the MRAM consumes the current shown in the standby specification of the datasheet.

Write time: During this phase, the non-volatile memory consumes the current shown in the active specification of the datasheet. Assuming a 3.3-V system with a tolerance of ±10%, the lowest voltage on the I/O could be 3.3 V – 10% = 2.97 V. This voltage of 2.97 V was used in calculations.

EST 2917   July, 2020

Energy Calculations

Energy used by an EEPROM

	Number of SPI bytes	Capacitance (µF)	Rise-time energy[3] (µJ)	Write time (ms)	Write energy[1] (µJ)	Total energy (µJ)	Energy/data byte[2] (µJ)
EEPROM	4	0.1	0.44	5.0	49.5	49.9	12.5
	46	0.1	0.44	5.0	49.5	49.9	1.0
Notes	1. Write energy is V_{DD} × Write current × time. 2. The number of data bytes is four less than the number of SPI bytes due to the required overhead. 3. Rise-time energy = 1/2 CV^2.						

TABLE 1: ENERGY CONSUMED DURING WRITE OPERATIONS FOR EEPROM

Let's start by looking at the energy used by the non-volatile memory during writing by a typical 3.3-V EEPROM with a standby current of 1 µA, a write time of 5 ms and a write current of 3 mA *(Table 1)*. We will assume that:

The EEPROM is ready to begin operation as soon as VDD rises to be within operating limits (a power up time of zero).

• The amount of data written fits into one page and writing takes place using the block write capability.
• The write time of the EEPROM is only that required to perform the write operation of the EEPROM, so we ignore any processing and communication time of the MCU and SPI interface. (This assumption is the opposite of that used for MRAM; MRAM only requires communication time because the write time is so short that it can be considered to be zero.)
• The EEPROM is powered directly from the microcontroller I/O and uses a small (0.1 µF) decoupling capacitor.

Energy used by a serial flash

Serial flash has much higher write and standby currents, so we will use a standby current of 50 µA, a write time of 3 ms, and a write current of 15 mA in our evaluation *(Table 2)*. As above, we assume zero power up time, the data fits into one page, and the write time is so long that we can ignore the communication time. In addition, we assume that the flash writes are to a pre-erased page.

TABLE 2: ENERGY CONSUMED DURING WRITE OPERATIONS FOR SERIAL FLASH

	Number of SPI bytes	Capacitance (µF)	Rise-time energy[3] (µJ)	Write time (ms)	Write energy[1] (µJ)	Total energy (µJ)	Energy/data byte[2] (µJ)
Flash	4	1.0	4.41	3.0	148.5	152.9	38.2
	46	1.0	4.41	3.0	148.5	152.9	3.3
Notes	1. Write energy is V_{DD} × Write current × time. 2. The number of data bytes is four less than the number of SPI bytes due to the required overhead. 3. Rise-time energy = 1/2 CV^2.						

Energy used by an MRAM

For MRAM, we based the evaluation on Everspin Technologies' MR25H256, a 256-kb serial SPI MRAM. *Table 3* shows the energy per data byte is lowest when using all of the energy from the decoupling capacitor. The decoupling-capacitor size should be chosen to match the amount of data that's typically acquired by the system.

TABLE 3: SERIAL MRAM ENERGY CONSUMED DURING WRITE OPERATIONS

Number of SPI bytes	Capacitance (µF)	Rise-time energy[4] (µJ)	t_{PU}	t_{PU} energy[1] (µJ)	Write time at 40 MHz (µs)	Write energy[2] (µJ)	Total energy (µJ)	Energy/data byte[3] (µJ)
4	1.0	4.41	400	0.14	1.6	0.06	4.6	1.15
46	1.0	4.41	400	0.14	10	0.48	5.0	0.11

Notes	1. The energy consumed by the MRAM during the power-up delay (t_{PU}) is given by V_{DD} × standby current × time. 2. Write energy is given by V_{DD} × Active current × time. 3. The number of data bytes is four less than the number of SPI bytes due to the required overhead. 4. Rise-time energy = 1/2 CV^2.

-µF capacitor allows the writing of 50 bytes (46 data bytes) on the SPI bus at 40 MHz with the MRAM consuming 27 mA. This calculation is the source of using 46 bytes for comparisons.

Energy used by the MCU and LDO

For the MCU, it might take 100 µs to wake up, make a measurement, and communicate the result to non-volatile memory and any required housekeeping. During this time, we assume an active current consumption of 500 µA (typical of small MCUs running at ~5 MHz). Thus, energy consumed is 3.3 V × 500 µA × 100 µs = 0.165 µJ per data acquisition.

In addition to the energy to make an acquisition, we should consider the energy required to keep the MCU active during the non-volatile memory write. When not acquiring or storing data, the MCU was in a sleep state that consumed 5 µA. The power supply was assumed to be an LDO that consumes 1 µA during all phases of operation (active and sleep).

Results summary

Table 4 shows the energy used by systems that make 10 and 100 acquisitions per second (four bytes per acquisition). For 100 acquisitions per second, EEPROM required 0.5 seconds just in write time. A larger number of acquisitions would have to look at caching the acquisition data and performing block writes. Figure 3 graphically summarizes the results.

Memory type	Number of acquisitions per second[4]	Active time[1] (ms)	Sleep time[2] (sec)	Energy consumed sleep time[3] (µJ)	Energy consumed making acquisitions (µJ)	Total energy (µJ)
MRAM	10	4.5	0.995	6.6	20.6	27.2
	100	4.5	0.955	6.3	206.4	212.6
EEPROM	10	5.05	0.950	6.3	93.4	99.6
	100	5.05	0.495	3.3	933.7	937.0
Serial flash	10	35.0	0.965	6.4	441.8	448.1
	100	35.0	0.650	4.3	4418.0	4422.3

TABLE 4: SUMMARY OF THE ENERGY CONSUMED PER ACQUISITION

Notes:
1. Active time is given as 100 µs of MCU time + storage time.
2. Sleep time is given as 1 – active time.
3. Energy consumed while asleep is given by $3.3\,V \times 2\,\mu A \times time$.
4. Four bytes per acquisition.

Total energy by type and number of acquisitions (μJ)
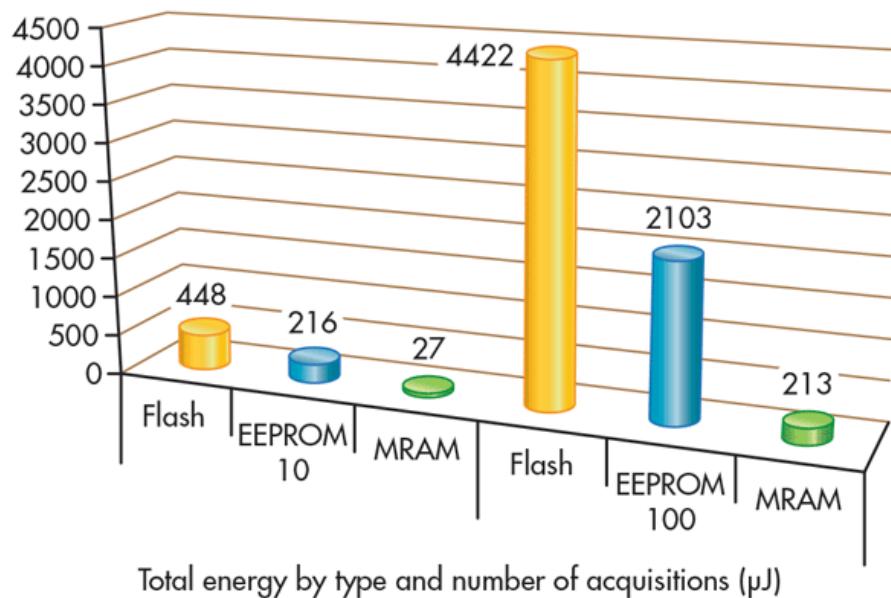
Energy Consumption Effect: Power Gating vs. Sleep Mode

EEPROM has low standby power consumption, so operating it with VDD always present could be considered. However, the write energy of just one write operation to EEPROM is equivalent to it being in standby mode for about 15 seconds. Again, the write energy is dominant.

Power gating really applies to MRAM, where the fast Write time significantly reduces the amount of energy used. The energy consumption of an acquisition using MRAM is 1.54 μJ. This is the same as a 3.3-V EEPROM with a standby current consumption of 1 μA being in standby for 0.46 seconds.

The write energy is dominant over standby energy.

Conclusion

The write time of a non-volatile memory greatly affects the total energy consumption of a system. The effects are less pronounced for systems with a low duty cycle, but become more pronounced with rising acquisition rates. The write time of EEPROM and flash significantly increase the energy consumption of the MCU, since they cause the MCU to be active for longer. Energy consumption could be reduced if the MCU was in a sleep mode while the writes to EEPROM and flash complete. However, the energy consumed by the EEPROM or flash represents most of the system's energy consumption, so having the MCU in sleep mode will not have a major impact on overall consumption.

It's clear that the lowest energy consumption can be achieved with a fast-write, non-volatile memory that's power gated.

*All credits go to Duncan Bennett, who has more than 27 years of experience in the electronics/semiconductor industry. He has spent the last 20 years in sales roles, including field applications, sales, area sales manager, strategic marketing, and product marketing and to Andrew Pockson, Divisional Marketing Manager at Anglia, who is responsible for wireless and semiconductor products. He has over 15 years of distribution experience in technical and marketing support roles (***Source URL:** http://electronicdesign.com/memory/mram-optimizes-system-energy-consumption)*

Contact Information:

Author:

Paolo Schiappacasse

FAE EMEA

Everspin Sales

How to Reach Us:

www.everspin.com

E-Mail:

support@everspin.com

orders@everspin.com

sales@everspin.com

USA/Canada/South and Central America

Everspin Technologies

5670 W. Chandler Road, Suite 100

Chandler, Arizona 85226

+1-877-347-MRAM (6726)

+1-480-347-1111

Europe, Middle East and Africa

support.europe@everspin.com

Japan

support.japan@everspin.com

Asia Pacific

support.asia@everspin.com

Everspin Technologies, Inc.