

# Flash raffiniert nutzen

Flash-Speicher bieten viel Kapazität zu niedrigen Kosten, doch für ihre Programmierung sind Flash-Controller mit ausgefeilten Algorithmen erforderlich. Eine neu vorgestellte Firmware kombiniert innovative Mapping-Verfahren mit vielen weiteren Ansätzen, um vor allem in Embedded-Anwendungen die Zuverlässigkeit der Speicher und die Robustheit gegen Spannungsausfälle zu steigern. So lässt sich entweder die Lebensdauer von SLC-Flash verlängern, oder der Einsatz von günstigerem MLC-Flash wird möglich.

AXEL MEHNERT

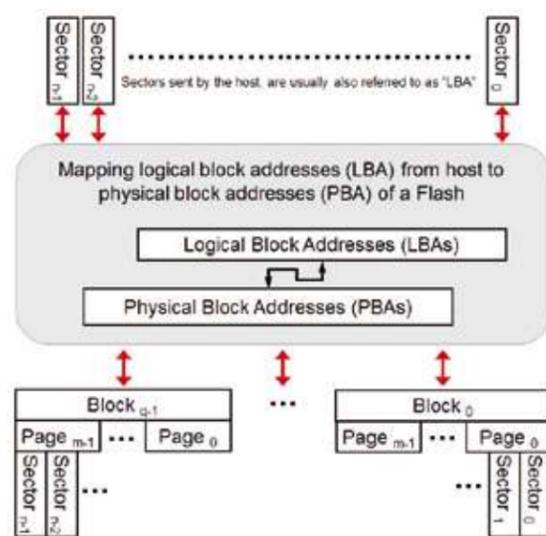
Flash-Speicher sind heute das Mittel der Wahl, wenn es um das Halten größerer Datenmengen bei schnellem Zugriff geht – sei es im Konsumelektronikbereich bei USB-Sticks oder »Solid State Disks« (SSD) oder auch im industriellen und Embedded-Sektor. Weil sich Flash-Speicher nur blockweise löschen und seitenweise beschreiben lassen, sind für ihre Ansteuerung Flash-Controller nötig, die sich vor allem durch die realisierte Firmware differenzieren und mit ausgefeilten Algorithmen ein detailliertes Management des physikalischen Flash-Speicherbereichs ermöglichen. Genau hier setzt Hyperstone mit seiner neuen »hyMap«-Firmware an und will gerade in Embedded- und industriellen Anwendungen die Zuverlässigkeit und Geschwindigkeit von Flash-Speicherlösungen deutlich verbessern.

Dateisysteme nutzen bestimmte Zugriffsmuster, um in einen Speicher zu schreiben. Die kleinste Zugriffseinheit ist übli-

cherweise ein Sektor von 512 Byte, 4 KByte sind ebenfalls relativ gebräuchlich, aber die Einheiten können auch durchaus 256 KByte groß sein. Die kleinste Einheit für den Lese-

zugriff auf einen NAND-Flash kann ein Sektor mit 512 Byte sein. Dagegen ist die kleinste Einheit für den Schreibzugriff eine Seite. Ein weiteres Flash-spezifisches Merkmal ist die Tatsache, dass ein ganzer Block gelöscht werden muss, bevor einzelne Seiten geschrieben werden können. Auch die Blockgrößen unterscheiden sich je nach Flash-Technologie und der Speicherkapazität.

Mehrere unterschiedliche Protokolle wie ATA, USB oder SD/MMC leiten die Lese- und Schreibzugriffe entweder direkt oder in einer Queue an den Controller des Speicherbausteins weiter. In allen Flash-Speichersystemen übernimmt ein »Flash Translation Layer« (FTL) das Mapping von logischen auf physikalische Adressen und übersetzt so die »logischen« Zugriffe auf der Host-Seite in »physikalische« Zugriffe auf der Flash-Seite. Hierbei kommen je nach Anforderungen der Anwendung unterschiedliche Mapping-Ansätze und Granularitäten zum Einsatz:



**Bild 1: Blockbasiertes Mapping bildet logische Blöcke auf physikalische Blöcke ab, und Seiten innerhalb der Blöcke folgen einer direkten 1:1-Allokation zwischen der Host- und der Flash-Sicht**

■ **Block Based Mapping (blockbasiertes Mapping)**

Einfaches blockbasiertes Mapping bildet logische Blöcke auf physikalische Blöcke ab, und Seiten innerhalb der Blöcke folgen einer direkten 1:1-Allokation zwischen der Host- und der Flash-Sicht (Bild 1).

■ **Page Based Mapping (seitenbasiertes Mapping)**

Beim seitenbasierten Mapping werden logische Seiten auf physikalische Seiten abgebildet, und die Blocknummern sind Teil einer Seitenadresse. Die logische Seite eins kann also auf jede beliebige Seite innerhalb jedes beliebigen Blocks abgebildet werden (Bild 2).

■ **Sub-Page-Based Mapping**

Wie der Name vermuten lässt, bildet »Sub-Page-Based Mapping« logische Einheiten, die kleiner sind als eine Seite, auf physikalische Einheiten ab (Bild 3). Weil der Flash-Speicher jedoch mindestens seitenweise geschrieben werden muss, konsolidiert das Verfahren diese Einheiten in einer Seite. Beim Lesen der kleineren Einheit muss der Controller nicht die gesamte Seite aus dem Flash in das SRAM des Controllers übernehmen. Die kleinste nutzbare Einheit ist ein Sektor oder wird durch die ECC-Blockgröße bestimmt.



**Dr. Jan Peter Berns**  
Geschäftsführer bei Hyperstone

Der Write Amplification Factor, kurz WAF, wird häufig im Bereich größerer Speichermedien wie SSDs für Enterprise-Server betrachtet. Dabei kommen überwiegend Controller mit externem DRAM zum Einsatz. Bei Wechselmedien, wie etwa USB, SD- oder CF-Karten, wurde dieser Faktor bisher nicht thematisiert. Die kostenoptimierten Controller für die Konsumelektronik bieten nicht genügend Speicherplatz und Prozessorleistung, um einen konkurrenzfähigen WAF zu ermöglichen und ein Sub-Page-basiertes Mapping anzubieten. Mit unserer neuen hyMap-Technologie sind wir nun in der Lage, auch für diese kleineren Formate kostengünstige MLC-Speicher einzusetzen bei vergleichbarer Lebensdauer und ohne Einschränkungen bei Zuverlässigkeit und Datensicherheit.

Die hyMap-Firmware ist von Haus aus Sub-Page-basiert und lässt sich auf verschiedene Granularitätsstufen hin kompilieren.

Mit der Granularität des Mappings steigt auch die Komplexität des Algorithmus. Nicht nur ist mehr Rechenleistung nötig, auch muss

## Intelligentes Design

beginnt mit Intelligent Analog PIC® Mikrocontrollern



Analogdesign ist schwer und erfordert wertvolle Entwicklungszeit. Microchips intelligente PIC® MCUs enthalten Analogfunktionen wie hochleistungsfähige A/D-Wandler, D/A-Wandler und Operationsverstärker und bieten somit eine einfache Schnittstelle zum Analogdesign. Eine Einchip-Lösung trägt somit zu geringeren Störungen im System bei und erhöht den Datendurchsatz – und das bei wesentlich kürzerer Entwicklungszeit und zu geringeren Kosten.

### Anwendungen

- Hochqualitative Umgebungssensoren
- Tragbare medizinische Geräte
- Industrieanlagen
- Leistungswandlung
- Effiziente Antriebe
- Beleuchtung
- Strommessung und -überwachung
- Energy Harvesting
- Solarwechselrichter



mehr Mapping-Information bei einem Update gespeichert und beim Lesezugriff wieder abgerufen werden. Für einen konventionellen seitenbasierten FTL fallen Mapping-Daten im Bereich von 0,1% der Laufwerkskapazität an, also beispielsweise etwa 32 MByte für einen 32-GByte-Speicher. Beim Sub-Page-Based-Mapping liegt dieser Anteil sogar noch höher. Die meisten SSDs (Solid State Drives) legen bisher die Zuordnungstabellen mit der Mapping-Information in einem externen DRAM ab. Jedes Mal, wenn ein Laufwerk herunterfährt, muss dieser DRAM-Inhalt in den nichtflüchtigen Speicher übertragen werden. Damit dieser Vorgang sicher stabil abläuft, übernehmen bei einem Stromausfall Supercaps die Spannungsversorgung. Diese Supercaps verursachen Kosten und reduzieren die Zuverlässigkeit des Laufwerks. Die Kondensatoren sind daher für viele Systeme keine Option. hyMap aktualisiert die Mapping-Information fortlaufend im nichtflüchtigen Speicher mithilfe eines ausgefeilten Algorithmus und eines transaktionsorientierten

		Block Based Mapping		Hyperstone hyMap	
		Read (MByte/s)	Write (MByte/s)	Read (MByte/s)	Write (MByte/s)
SLC	Sequenziell	95	60	90	60
	4K Random	5,7	0,09 (20 IOPS)	3,1	4,0 (1000 IOPS)
MLC	Sequenziell	75	25	70	25
	4K Random	4,7	0,03 (7 IOPS)	3,3	3,9 (975 IOPS)

Tabelle 1: Geschwindigkeitsvergleich von Block-Based-Mapping und hyMap für eine SD-Karte im Modus »SD 3.0 UHS-I« mit S8 (SLC: 2x nm, 8K Page, 100 MHz DDR I/F, 2 CE DDP; MLC: 1x nm, 16K Page, 100 MHz DDR I/F, 2 CE DDP, IOPS: I/O-Operation pro Sekunde)

zahl in den Flash geschriebene Bytes geteilt durch Anzahl vom Host geschriebene Bytes). So bedeutet beispielsweise ein WAF von 4, bezogen auf zufällige Schreibvorgänge von 4 KByte, dass für jeden 4-KByte-Datensatz tatsächlich 16 KByte in den Flash-Speicher geschrieben werden. Und auch wenn Blöcke gelöscht werden müssen, um Kapazität freizugeben und ungenutzte Seiten zu konsolidieren, die über verschiedene Blöcke verteilt

- Mapping-Granularität,
- Effizienz des Mappings und der Garbage Collection,
- Flash-Technologie (SLC/MLC, Blockgröße, Seitengröße),
- Daten-Caching,
- Speicherort für Mappingdaten (externes DRAM oder Flash),
- Befehls-Queuing, falls vom Protokoll erlaubt,

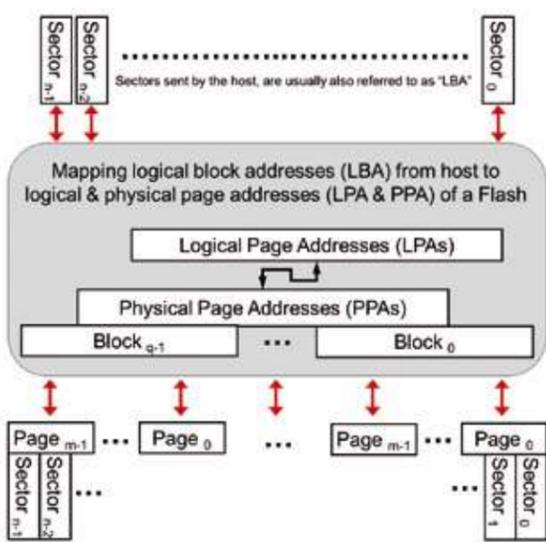


Bild 2: Seitenbasiertes Mapping bildet logische Seiten auf physikalische Seiten ab, die logische Seite eins kann also auf jede beliebige Seite innerhalb jedes beliebigen Blocks abgebildet werden

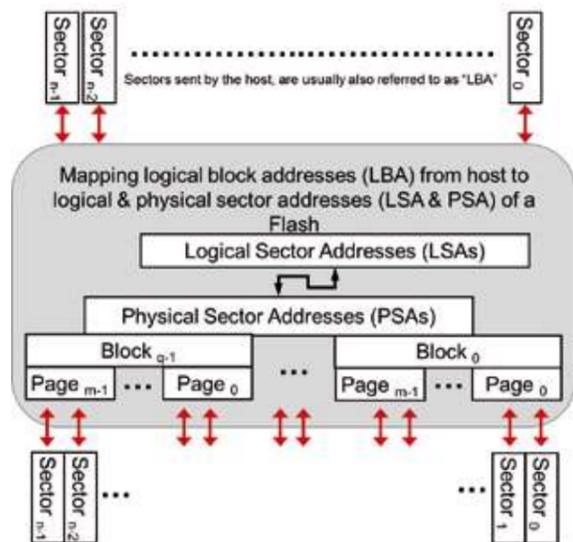


Bild 3: Prinzip des »Sub-Page Based Mapping«, bei dem einzelne Sektoren auf verschiedenen beliebigen Seiten und Blöcken verteilt sein können

Logbuchs. Dadurch ist die Datensicherheit bei Ausfall der Versorgungsspannung zu jedem Zeitpunkt gewährleistet.

### Write Amplification

Der »Write Amplification Factor« (WAF) gibt die Informationsmenge an, die zusammen mit einer bestimmten Menge von Anwenderdaten zusätzlich in einen Flash-Speicher geschrieben werden muss (WAF = An-

sind, kann dies bedeuten, dass ein 4-KByte-Schreibzugriff in einem Blocklöschvorgang und dem Schreiben von mehreren Seiten in diesem Block resultiert. Für blockbasiertes Mapping könnte ein WAF im Bereich von 30 bis 200 für zufällige 4-KByte-Schreibzugriffe als realistische Schätzung gelten. Auf der anderen Seite kann der WAF für rein sequenzielle Schreibvorgänge und größere Datenmengen durchaus auch unter drei liegen.

Den WAF zu berechnen ist komplex und hängt von verschiedenen Faktoren ab wie:

- Datenzugriffs-/schreibmuster,
- freie Kapazität auf dem Laufwerk und Organisation dieses freien Anwenderbereichs (TRIM),
- Überdimensionierung,
- Datenintegritätsanforderungen des Hosts oder Dateisystems und Möglichkeit, Command-Queuing und Caching von Schreibdaten zu nutzen, um den WAF zu reduzieren.

Für blockbasiertes Mapping kann der WAF speziell bei kleinen und zufälligen

Schreibzugriffen signifikant sein. Ein Einzelsektor-Schreibzugriff kann unter Worst-Case-Bedingungen erforderlich machen, dass ein gesamter Block gelöscht wird, dessen vollständiger Originalinhalt kopiert und zusammen mit dem einzelnen aktualisierten Sektor in diesem Block abgespeichert wird (Bild 4). In einem solchen Szenario läge der WAF bei 2000, da es für einen 512-Byte-Sektor erforderlich ist, einen ganzen Block von 1 MByte zu schreiben.

Beim Sub-Page-basierten Mapping wird dieser Worst-Case-Faktor deutlich reduziert. Sollte kein ungenutzter Block verfügbar sein, kann für einen Einzelsektor-Schreibzugriff unter Worst-Case-Bedingungen zwar auch hier ein gesamter Block gelöscht werden müssen, aber es müssten keine bestehenden Daten kopiert werden. Es würden in den Tabellen nur der bestehende Sektor ungültig und der neue Sektor in einem anderen Block gültig markiert. Um einen Einzelsektor zu programmieren, könnte es im ungünstigsten Fall nötig sein, die ganze Seite zu programmieren. In diesem Fall (Bild 5) läge der WAF bei 16 (8 KByte / 512 Byte).

Der Vergleich beider Mapping-Ansätze zeigt, dass ein feineres Mapping in diesem

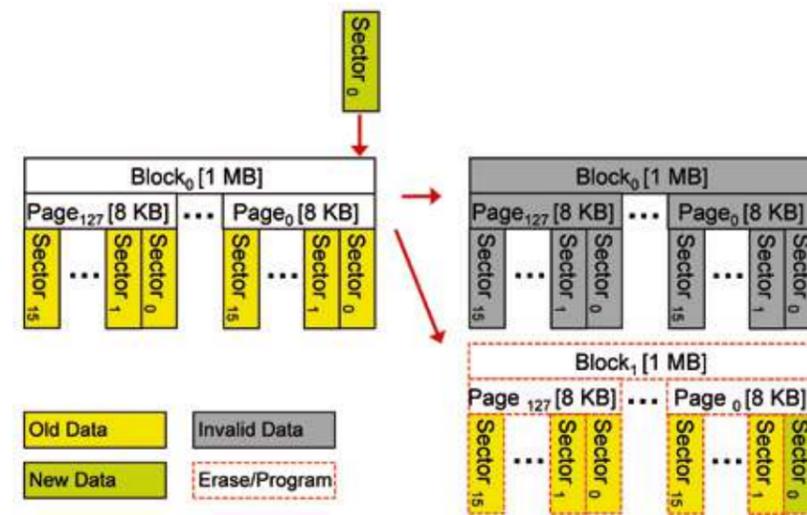


Bild 4: Für einen zufälligen Einzelsektor-Schreibzugriff bei blockbasiertem Mapping muss im Worst Case ein gesamter Block (gestrichelt) neu geschrieben werden – ein Schreibfaktor (WAF) von 2000!

Szenario deutliche Vorteile bringt und die Lebensdauer des Laufwerks fast um den Faktor 100 steigert. Zwar gibt es in den meisten Anwendungsfällen nicht nur Einzelsektor-Schreibzugriffe, und auch der Controller kann sehr oft mehrere Sektoren auf einmal in den Flash schreiben. Leider kennen die meisten

Anwender jedoch die genauen Zugriffsmuster ihrer Hosts gar nicht. Für Nutzungsmodelle mit einem Mix von Nutzlastgrößen ist es recht schwierig, genaue WAF-Werte zu berechnen – aber als Größenordnung dürfte der WAF von Sub-Page-basiertem Mapping mehr als 30-fach besser sein.

## embedded world 24.-26. Februar

Don't miss  
the future!

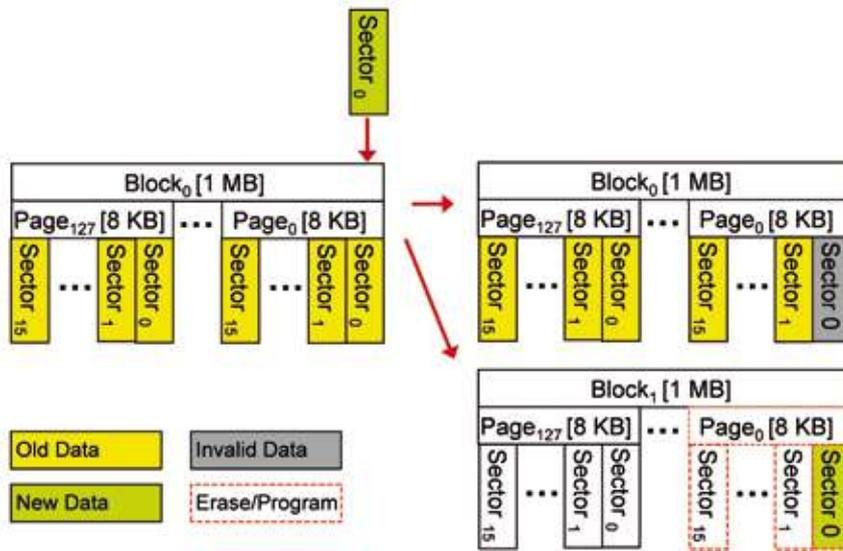
- ✓ Täglich Live-Demos aus den Bereichen IoT, Messtechnik und Bauelemente
- ✓ Erleben Sie die neuesten Entwicklungskits führender Hersteller
- ✓ Gewinnen Sie einen 3D-Scanner

Explore  
the Future  
Hall 5, Stand 318

Besuchen Sie uns!

Mehr Infos und kostenfreie Tickets unter:  
embeddedworld.conrad.de

CONRAD  
Business Supplies



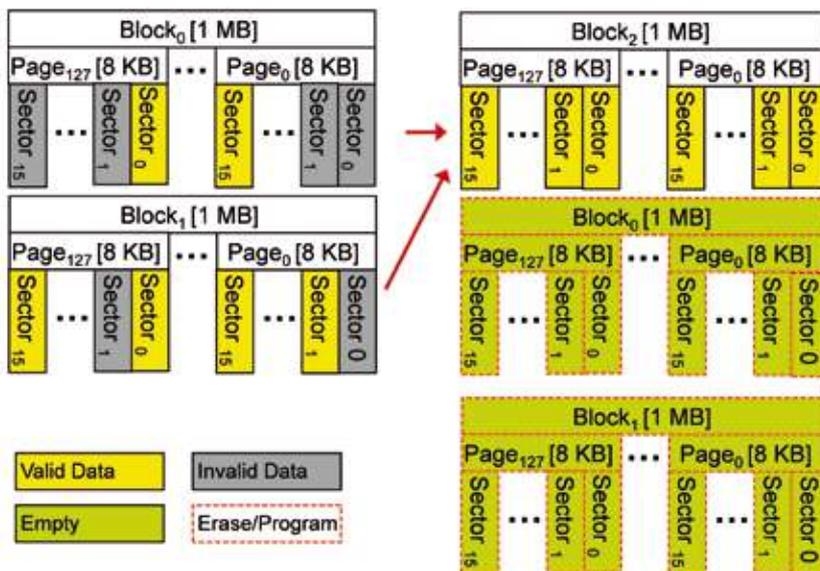
**Bild 5:** Bei seitensbasiertem Mapping liegt der Worst-Case-WAF für den zufälligen Einzelsektor-Schreibzugriff nur bei 16, da nur eine Speicherseite (gestrichelt) neu geschrieben werden muss

Neben der Kapazität, der Flash-Technologie und dem Nutzungsmodell ist der Write-Amplification-Faktor ein wichtiges Element für die Abschätzung der Lebensdauer des Laufwerks. Wenn der Entwickler den WAF in Verbindung mit dem eigenen Nutzungsmodell und die Vorgeschichte des Laufwerks kennt, kann er die gesamte Standzeit des Laufwerks ermitteln, ausgedrückt in »Tera Bytes Written« (TBW), und die Laufwerke spezifizieren:  $TBW = \text{Kapazität [GByte]} / 1000 * PE\text{-Zyklen} / WAF$ . Legt man ein Szenario zugrunde, bei dem der WAF eines individuellen Nutzungsmodells im Bereich von 5 für hyMap liegt und bei 100 für eine blockbasierte Mapping-Architektur, kann es sich lohnen, ein

SLC-Flash mit 100 000 Schreib-/Löschzyklen durch ein günstigeres MLC-Flash zu ersetzen, das 5000 Schreib-/Löschzyklen bietet, und dabei dieselbe Lebensdauer in Bezug auf »Terabytes Written« (TBW) zu bekommen.

### Garbage Collection

Bei hyMap läuft die »Garbage Collection«, also das Zusammenfassen physikalischer Blöcke, die Sektoren oder Seiten mit abgelaufenen Daten enthalten, um Speicherplatz wieder freizugeben, soweit wie möglich im Hintergrund und zu Zeiten ab, die sonst Leerlauf aufweisen (Bild 6). Um die Auswir-



**Bild 6:** Bei der »Garbage Collection« werden physikalische Blöcke zusammengefasst, die Sektoren oder Seiten mit abgelaufenen Daten enthalten, um Speicherplatz wieder freizugeben

kung der Garbage Collection auf die Speichergeschwindigkeit klein zu halten, lassen sich ausreichende Überkapazitäten auf dem Flash bereitstellen. Unter diesem »Over-Provisioning« ist die Verfügbarkeit überzähliger physikalischer Blöcke nichtflüchtigen Speichers im Vergleich mit der Nettokapazität des logischen Datenraums zu verstehen. Je nach Dateisystem kann hyMap nicht verwendete Kapazität als dynamische Überkapazität nutzen, und der Algorithmus verwendet die gesamte verfügbare Überkapazität, um den WAF und die Geschwindigkeit zu optimieren. Wie sich die Summe der Maßnahmen (Sub-Page-basiertes Mapping, Garbage Collecting usw.) auf die Geschwindigkeit der Speicher auswirkt, zeigt Tabelle 1. Um die optimale Lebensdauer zu gewährleisten, nutzt hyMap die hocheffizienten Wear-Levelling-Algorithmen der bisherigen Hyperstone-Firmware.

Im Falle eines Spannungseinbruchs erfolgt ein Controller-Reset, und der Schreibschutz des Flash wird aktiviert. Die Firmware protokolliert sämtliche Flash-Transaktionen

**AXEL MEHNERT**



**ist Vice President Marketing & Product Strategy bei Hyperstone**

– sollten sich die jüngsten Daten als defekt erweisen, wird der Controller den letzten gültigen Eintrag vor dem letzten missglückten Schreibvorgang wiederherstellen. Und auch MLC-Flash so sicher wie möglich zu machen, ist ein Ziel von hyMap: Da zwei logische MLC-Flash-Seiten korreliert sind, ist es möglich, Daten einer älteren Seite zu zerstören, wenn eine andere neue Seite innerhalb desselben Blocks geschrieben wird. hyMap wendet eine »Reliable Write«-Technik an, damit MLC bei Spannungsausfall sicher bleibt. Darüber hinaus sorgt »Safe Flash Handling« dafür, auch mit schwach programmierten Daten im Flash umzugehen. hyMap wird ausführlich während Flash-Management-Aktivitäten und mit kritischem Timing getestet, um maximale Datensicherheit bei Spannungsausfällen zu gewährleisten. Auch hier kann hyMap Geld sparen, da es die Möglichkeit eröffnet, MLC zu nutzen oder in SLC-basierten Systemen mehr Lebensdauer in TBW zu erhalten. (cg)

**Hyperstone**  
 Telefon: 0 75 31/98 03 0  
 www.hyperstone.com

**H1  
 301**