

Speichertechnik

Einfach **fallen** lassen

Pseudostatische Speicherchips sind so konzipiert, dass sie in die Fassung von statischem RAM eingesteckt werden können, obwohl die interne Arbeitsweise des Speichers nichtstatischer Natur ist. Zwei gängige pseudostatische Speichertypen sind PSRAM und FRAM. Wie gelingt es nun diesen pseudostatischen Speichertypen, mit ihrer Maskerade Erfolg zu haben? Was hat der Systementwickler davon?

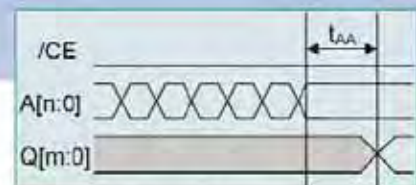


Bild 1: ATD bei SRAM

Jarrod Eliason

PSRAM (Pseudostatisches RAM) zielt auf langsame SRAM-Anwendungen ab und konkurriert auf reiner Kosten-pro-Bit-Basis. FRAM (Ferroelektrisches RAM) hat Anwendungen mit batteriegepuffertem SRAM (BBSRAM) im Visier und konkurriert auf der Basis von Systemkosten und Logistik. FRAM zielt außerdem auf nichtflüchtige Datenerfassung ab, wo dieser Speichertyp seine besonderen Stärken ausspielt.

Die Auswertung von Änderungen auf den Adressleitungen (ATD - Address Transition Detection) ist eine besondere Eigenschaft von asynchronem SRAM. Diese Speicherchips gestatten es den Adresspins, sich fortgesetzt beliebig schnell und beliebig lang zu ändern. Dabei gewährleistet der Speicher die Ausgabe der korrekten Daten innerhalb der Zeit t_{AA} (Adress-Zugriffszeit), in der sich die

Adresspins stabilisieren (siehe Bild 1). Diese Flexibilität gibt den Entwicklern von Mikrocontrollern und Mikroprozessoren viel Freiheit bei der Steuerung des relativen Timings der Signale an den Speicherschnittstellen. Bei einem typischen Controller mit direktem Speicherinterface (d.h. ohne Bündelung von Adressen und Daten) werden Chipselect-Output und Adressen von derselben Taktflanke gesteuert (siehe Bild 2). Während das Timing zwischen »CLK« und »/CS« (tCS) bzw. »CLK« und »A« (tADR) im Datenblatt normalerweise angegeben ist, findet das relative Timing zwischen /CS und A gewöhnlich nicht einmal Erwähnung und ist schon gar nicht garantiert. Bei SRAM spielt das keine Rolle. Die einzige Voraussetzung besteht darin, dass die Ad-

ress-Zugriffszeit kurz genug ist. Die vom Controller erlaubte Zeit beträgt zwei Taktperioden (T) minus Adress- oder Chipselect-Laufzeitverzögerung (Maximum von tADR und tCS) und Daten-Setupzeit des Controllers (tSU). In den meisten Fällen haben tADR und tCS den gleichen Wert. Für beide Größen laufen die Berechnungen aufs Gleiche hinaus. Um die Anzahl der Pins in Grenzen zu halten, erfüllt fast jeder Pin an einem modernen Controller mehrere Zwecke. Eine Nebenwirkung dieser Funktionsbündelung besteht

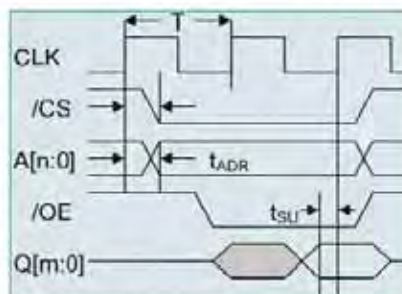


Bild 2: Direktes Controllerinterface

darin, dass jeder Pin eine andere interne Laufzeitverzögerung haben kann. Wenn nicht jede Verzögerung für sich kontrolliert wird, ist es wahrscheinlich, dass wenigstens ein Adresspin etwas langsamer als Chipselect ist. Erschwerend kommen auch externe Faktoren hinzu: Es ist gängige Systemdesign-Technik, den Adressbus von Speicher und/oder Peripherie gemeinsam nutzen zu lassen, während Peripheriekomponenten die Chipselects normalerweise nicht gemeinsam nutzen. Auch wenn die Adressen dem vom Controller kommenden Chipselect-Signal vorausgehen, kann diese Timing-Beziehung an den Pins der Speicherkomponente verloren gegangen sein. Ein asynchrones SRAM betrachtet einen Address-Skew (Versatz) als Verringerung der erforderlichen Zugriffszeit, was die Funktionsweise je-

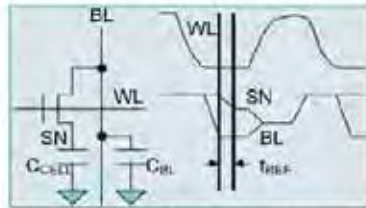


Bild 3: DRAM-Write/Read einer »1«

doch nicht berührt. Für pseudostatische Speicher ist das relative Timing zwischen Chipselect und langsamster Adresse allerdings ein kritischer Parameter, wie gleich gezeigt werden soll.

Relativ pünktlich

Intern arbeitet PSRAM wie dynamisches RAM (DRAM) in dem Sinne, dass der Speicher laufend im Hintergrund aufgefrischt werden muss, um den Speicherzustand aufrecht zu erhalten. Des Weiteren beeinträchtigt der Vorgang des Lesens einer PSRAM-Zelle das Signal in dieser Zelle, und es dauert seine Zeit, den Datenzustand abzutasten und das Signal wieder auf den vollen Pegel zurückzusetzen. Wie in Bild 3 vereinfacht dargestellt, speichert eine DRAM-Zelle Daten durch Festhalten der Ladung auf dem Speicherknoten (SN) des Zellenkondensators (C_{CELL}).

Ein Transistor dient als Schalter, um den Zugriff auf die Zelle zu kontrollieren. Viele Zellen teilen sich dieselbe Bitleitung, und die parasitische Kapazität jedes Transistor-Drains zusätzlich zur parasitischen Metallkapazität bewirkt eine Nettokapazität, die in Bild 3 als C_{BL} bezeichnet ist. Die in der Zelle gespeicherte Ladung fließt im Laufe der Zeit langsam über den Zugriffstransistor und Zellenkondensator ab. Die Spannung am Speicherknoten muss aufgefrischt werden,

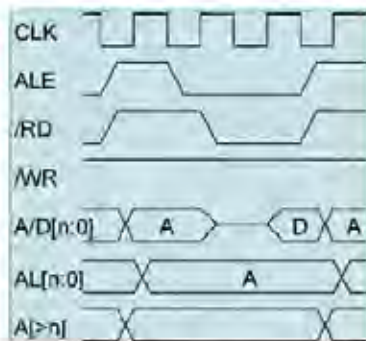


Bild 4: Gemultiplexte Schnittstelle

bevor das Signal unter den erfassbaren Pegel absinkt. Diese Auffrischzeit ist in der Abbildung als t_{REF} bezeichnet. Beim Auslesen oder Auffrischen der Zelle steigt die Spannung in der Wortleitung (WL). Die im Speicherknoten verbleibende Ladung verteilt sich auf C_{CELL} und C_{BL} . Dann wird die Spannung auf der Bitleitung abgetastet, um zu bestimmen, ob die Zelle einen Datenzustand von 1 oder 0 enthielt. Darauf findet die volle Wiederherstellung der mit diesem Datenzustand assoziierten Spannung auf dem Spannungsknoten statt.

Nach der Ladungsverteilung und vor der Wiederherstellung kann die auf dem Speicherknoten verbleibende Spannung unter den abtastbaren Pegel fallen. Wird der Zugriff innerhalb dieses kritischen Zeitfensters abgebrochen, könnte der nachfolgende Zugriff auf das gleiche Bit falsch abgetastet werden. Daher gilt der Vorgang

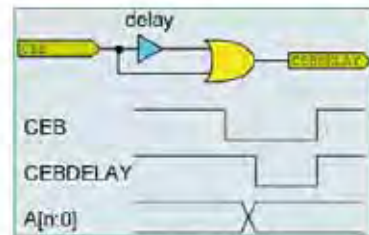


Bild 5: Verzögerung an der fallenden Flanke

des Auslesens eines DRAMs als destruktiv. Hat der Zugriff auf ein DRAM erst einmal begonnen, muss er vollständig zu Ende geführt werden. Ein DRAM ist nicht in der Lage, das in Bild 1 dargestellte Szenario laufender Adressänderungen so zu bewältigen wie ein SRAM. Allerdings kann ein DRAM so konzipiert sein, dass es die endlichen Timing-Unsicherheiten bewältigt, die in den meisten Controller-basierten Systemen vorkommen. FRAM ist von Natur aus nichtflüchtig und erfordert kein ständiges Auffrischen, jedoch gibt es hier – wie bei DRAM – destruktives Lesen, und es lassen sich gängige Techniken einsetzen, um einfach zu verwendendes pseudostatisches FRAM zu bilden.

Bild 4 zeigt einen weiteren gern verwendeten Weg, wie Controller die ATD-Eigenschaft von SRAM nutzen. Viele Controller bündeln die Adress- und Datenleitungen, um die Anzahl der Pins zu reduzieren, die für die Im-

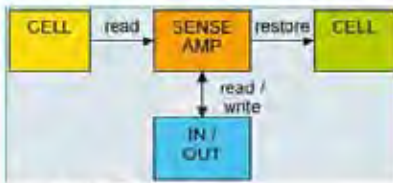


Bild 6: FRAM-Zugriff

plementierung der externen Speicherschnittstelle erforderlich sind. Zwischen Controller und Speicher ist ein externes Latch (Auffangregister) einzufügen, und der Controller liefert ein ALE-Signal (Address Latch-Enable) zur Kontrolle dieses externen Auffangregisters. Wenn ALE abfällt, werden die Daten am Input des Latches (A/D[n:0]) zwischengespeichert und während des gesamten Speicherzugriffs an den Speicherengängen (AL[n:0]) erhalten. Sobald die Adresse zwischengespeichert ist, geht der Controller vom A/D-Bus weg und setzt das /RD-Signal bei einem Lesevorgang auf »Low«, wie in Bild 4 gezeigt. Im Fall eines Schreibvorgangs legt der Controller die Daten auf den A/D-Bus und setzt das /WR-Signal auf Low. Nicht alle Adressen in Bild 4 sind gelatcht, und die Laufzeitverzögerung des Latches fügt bei den Adressen einen Skew hinzu.

Viele mit dem Adress-Latch arbeitende Controller liefern kein dediziertes Chip-Enable-Signal. In diesem Szenario können obere Adressen mit Glue-Logic zum Einsatz kommen, um zwischen verschiedenen Speicherchips oder anderen Peripheriegeräten zu wählen. Der /CE-Pin des Speicherchips kann während der Gesamtdauer

des Speicherzugriffs Low bleiben, und die Adressenänderungen sind der einzige Hinweis darauf, dass ein Zugriff beendet ist und ein weiterer beginnt.

Damit ein pseudostatischer Speicherchip in einen SRAM-Sockel seinen Dienst tun kann, muss er die folgenden Address-Skew-Szenarien berücksichtigen:

- Address-Skew von Chipselect (negative Adressen-Setup-Zeit) und
- Address-Skew von anderen Adressen bei ATD-Zugriffen.

Das erste Szenario lässt sich leicht über längere Zugriffs- und Zykluszeiten bewältigen. Die Lösung für das zweite Szenario ergibt sich ganz von selbst als Folge der pseudostatischen ATD-Implementierung. Bild 5 zeigt eine einfache asynchrone Verzögerungsschaltung, die dazu dienen kann, intern die abfallende Flanke des Chip-Enable-Signals zu verzögern, damit nach dem Abfallen des Chip-Enable-Signals für die Adressenänderungen noch Zeit bleibt.

Maskerade

Zwar spezifizieren die Controller-Datenblätter die Größenordnung der erforderlichen negativen Setup-Zeit nur selten, so dass sie eigentlich für jedes System individuell charakterisiert werden müsste, in den allermeisten Designs liegt sie jedoch unter 5 ns. Geht man von diesem

Wert aus, so besteht der Nachteil bei diesem einfachen Ansatz darin, dass er den Beginn des Speicherzugriffs verzögert und daher unmittelbar die Zugriffszeit verlängert. Die Verzögerung, die erforderlich ist, um unter schnellen Bedingungen eine Adressen-Setupzeit von -5 ns zu garantieren, kann die Zugriffszeit unter langsamen Bedingungen um bis zu 12 ns verlängern. Entsprechend können Lieferanten von pseudostatischem Speicher unterschiedliche Geschwindigkeitsgrade anbieten – je nachdem, ob die Verzögerung an der fallenden Flanke des Chip-Enable-Signals freigegeben ist oder nicht. So arbeitet der parallele 4-Megabit-FRAM-Speicherchip »FM22 L16« von Ramtron normalerweise mit einer Zugriffszeit von 55 ns, wenn eine Adressen-Setupzeit von 0 ns garantiert werden soll. Auf Anfrage lässt sich eine testprogrammierbare Option aktivieren, welche die Adressen-Setupzeit auf -5 ns setzt, wobei sich als Nebenwirkung die Zugriffszeit allerdings auf 70 ns erhöht.

Tatsächlich ist die pseudostatische ATD-Implementierung viel einfacher und sicherer als die komplett asynchrone in SRAMs. Jeder Entwickler von Speicherchips kennt wenigstens eine Horrorgeschichte über eine fehlerhafte ATD-Implementierung. Da jeder in einem pseu-

dostatischen Speicher gestartete Zugriff auch beendet werden und die Voraussetzung eines negativen Adressen-Setups erfüllt sein muss, besteht ATD nur noch in einem einfachen Vergleich der Adressen an den Inputs, wenn der interne Lesevorgang abgeschlossen ist, mit den Adressen, die intern zu Beginn des Zugriffs gelatcht wurden.

In einem FRAM-Chip ist der interne Zugriff in eine Lese- und eine Rückschreibphase aufgeteilt. Dies geschieht aus zwei Gründen. Erstens ist da die SRAM-Forderung auf Kompatibilität mit spätem Schreiben, die mit der Möglichkeit zu tun hat, dass der /WE-Input erst nach /CE fällt oder sich Daten ändern, nachdem /WE gefallen ist. Zweitens ist der interne Datenbus gewöhnlich breiter als der externe Bus. Bei einem internen Bus von 64 Bit und einem externen Bus von 16 Bit gibt es auch bei Schreibzyklen 48 zusätzliche Bits, die zu lesen und wiederherzustellen sind. Dazu beginnen und arbeiten alle FRAM-Zugriffe als Lesevor-



Bild 7: ATD pseudostatisch

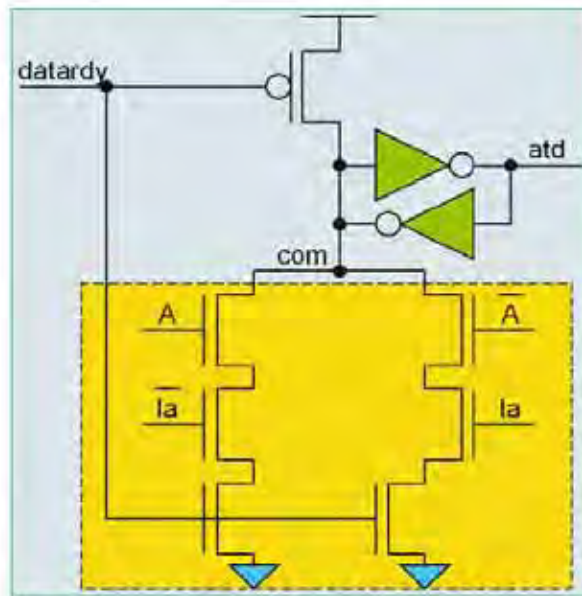


Bild 8: Adressenvergleichsschaltung

gang bis zu dem Punkt, an dem die von den internen Sensorverstärkern gelatchten Daten mit dem externen Datenpfad verbunden sind. Bei einem Lesevorgang werden die Sense-Amp-Daten auf die Datenpins des Speicherchips geleitet. Bei einem Schreibvorgang laufen die Daten von den Datenpins des Speicherchips in den Sense-Amp. Die Richtung des Datenflusses ist demnach der einzige Unterschied zwischen Lesen und Schreiben. Der Wiederherstellungs- bzw. Rückschreibvorgang arbeitet in beiden Fällen in gleicher Weise, wobei die im Sense-Amp gelatchten Daten in die Zelle zurückfließen. Bild 6 illustriert den Datenfluss bei einem FRAM-Zugriff. In älteren FRAM-Speicherchips steuerte allein der /CE-Pin den Übergang zwischen Phasen. Die abfallende Flanke des /CE-Signals leitete die Lese- und Schreibphase ein, während die ansteigende Flanke von /CE die Rückschreibphase startete. Als ATD hinzukam, wurde es erforderlich, dass Adressen auch in der Lage sind, Übergänge zwischen Phasen zu in-

itieren. Bild 7 zeigt die Modifikation der internen Version des Chip-Enable-Signals (cebint), um ATD Rechnung zu tragen. In Bild 7 erscheinen externe Signale in Großbuchstaben, während interne Signale klein geschrieben sind. Das interne cebint-Signal ist – wie bereits erklärt – gegenüber dem externen /CE-Signal geringfügig verzögert. Sobald die Lese- und Schreibphase des Zugriffs beginnt, werden die Adressen intern gelatcht. Das Latch-Kontrollsignal ist als alatch und die gelatchten Adressen als la[n:0] bezeichnet. Vor Abschluss der Lese- und Schreibphase wird jede Änderung in externen Adressen ignoriert. Ist die Lese- und Schreibphase abgeschlossen – wie vom datardy-Signal (Data Ready) angezeigt – geht das atd-Signal auf High, sobald der externe A[n:0]-Bus vom internen la[n:0]-Bus abweicht. In Bild 7 hat sich die externe Adresse vor dem Anstieg von datardy geändert, so dass das atd-Signal unmittelbar nach datardy auf High geht. Sobald das atd-Signal einen Adressenübergang anzeigt,

wird das interne cebint-Signal auf High gesetzt, um die Rückschreibphase zu starten. Wenn die Rückschreibphase abgeschlossen ist, geht die Kontrolle von cebint an den externen /CE-Pin zurück. Falls /CE immer noch Low ist, beginnt ein neuer Zugriff auf die bei Abschluss der Rückschreibphase aktuelle Adresse. Bei dieser Implementierung dient die Memory-Rückschreibzeit dazu, auch die Address-Skew-Zeit unterzubringen. Die schnellste Adresse initiiert ATD, und die langsamste Adresse hat die gesamte Speicher-Rückschreibzeit, um anzukommen. Die meisten pseudostatischen Speicherchips tolerieren Skew-Zeiten von Adresse zu Adresse von mehr als 10 ns und sogar bis 20 ns.

Die Adressenvergleichsschaltung selbst ist sehr einfach, wie Bild 8 zeigt. Der Einfachheit halber ist hier nur ein Bit des Adressenvergleichs dargestellt. Für jedes Adressenbit wiederholt sich die Schaltung innerhalb des gestrichelten Umrisses. Das gemeinsame Vergleichssignal (com) ist auf High vorge-laden, und der Entladungspfad wird unterbrochen, wenn das datardy-Signal auf Low steht. Wenn datardy auf High geht, wird der Entladungspfad freigegeben. Sind A und la jedoch gleich, ist je ein NMOS-Transistor auf jeder Seite ausgeschaltet und das com-Signal bleibt High, während das atd-Signal Low bleibt. Wenn A und la unterschiedlich werden, sind alle drei NMOS-Transistoren auf einer Seite eingeschaltet und entladen den com-Knoten, was wiederum das atd-Signal auf High latcht. (mc)

Jarrod Eliason
 ist Senior Design Engineer bei
Ramtron International
 Telefon 00 44/13 44 39 27 85
 www.ramtron.com