

άλλος επεξεργαστής θα ζητήσει την ίδια γραμμή.

Στο πρωτόκολλο ανανέωσης ενημέρωσης εγγραφής, μπορεί να υπάρχουν πολλοί επεξεργαστές καθώς και πολλοί αναγνώστες. Όταν ένας επεξεργαστής επιθυμεί να ανανεώσει ενημερώσει μια διαμοιραζόμενη γραμμή, η λέξη που πρόκειται να ανανεωθεί κατανέμεται σε όλους τους άλλους, και οι μνήμες cache που περιέχουν την γραμμή αυτή μπορούν να την ανανεώσουν.

Σε κάθε περίπτωση, καμία από τις δύο αυτές προσεγγίσεις δεν είναι ανώτερη από την άλλη. Η λειτουργία εξαρτάται από το πλήθος των τοπικών μνημών cache και από τον τρόπο με τον οποίο πραγματοποιούνται οι αναγνώσεις και οι εγγραφές μνήμης. Μερικά συστήματα υλοποιούν προσαρμοστικά πρωτόκολλα που χρησιμοποιούν και τους δύο μηχανισμούς ακύρωσης εγγραφής και ανανέωσης ενημέρωσης εγγραφής.

Η προσέγγιση ακύρωσης εγγραφής χρησιμοποιείται περισσότερο στα συστήματα πολυεπεξεργαστών του εμπορίου, όπως είναι οι Pentium 4 και PowerPC. Σημαίνει την κατάσταση κάθε γραμμής μνήμης cache (με δύο επιπλέον bit στην ετικέτα της μνήμης cache) με την ένδειξη τροποποιημένη, αποκλειστική, διαμοιραζόμενη, ή άκυρη. Για την αιτία αυτή, το πρωτόκολλο ακύρωσης εγγραφής ονομάζεται MESI (modified exclusive, shared, invalid: τροποποιημένη, αποκλειστική, διαμοιραζόμενη, άκυρη). Στη υπολοιπή ενότητα, θα δούμε την χρήση του μεταξύ τοπικών μνημών cache κατά μήκος πολυεπεξεργαστή. Για απλούστευση της παρουσίασης, δεν θα εξετάσουμε τους μηχανισμούς συντονισμού μεταξύ επιπέδου 1 και επιπέδου 2 τοπικά καθώς και του ταυτόχρονου συντονισμού κατά μήκος του καταναμιημένου πολυεπεξεργαστή. Μια τέτοια εξέταση δεν θα προσέθετε νέες αρχές αλλά θα περιέπλεκε την εξέταση σε μεγάλο βαθμό.

### Το Πρωτόκολλο MESI

Για να υπάρξει συμφωνία μνημών cache σε ένα SMP, συχνά η μνήμη cache δεδομένων υποστηρίζει ένα πρωτόκολλο που είναι γνωστό με το όνομα MESI. Στο MESI, η μνήμη cache δεδομένων περιλαμβάνει δύο bit κατάστασης ανά ετικέτα, έτσι ώστε κάθε γραμμή να μπορεί να βρίσκεται σε μια από τέσσερις καταστάσεις:

- **Τροποποιημένη:** Η γραμμή στην μνήμη cache έχει τροποποιηθεί (είναι διαφορετική από την κύρια μνήμη) και είναι διαθέσιμη μόνο σ' αυτή την μνήμη cache.
- **Αποκλειστική:** Η γραμμή στην μνήμη cache είναι ίδια με την γραμμή στην κύρια μνήμη και δεν υπάρχει σε καμία άλλη μνήμη cache.
- **Διαμοιραζόμενη:** Η γραμμή στην μνήμη cache είναι ίδια με την γραμμή στην κύρια μνήμη και μπορεί να υπάρχει και σε άλλη μνήμη cache.
- **Άκυρη:** Η γραμμή στην μνήμη cache δεν περιέχει έγκυρα δεδομένα.

Στον Πίνακα 18.2 δίνεται περιληπτικά η σημασία των τεσσάρων καταστάσεων. Στο Σχήμα 18.8 φαίνεται ένα καταστατικό διάγραμμα για το πρωτόκολλο MESI. Πρέπει να θυμόμαστε ότι κάθε γραμμή της μνήμης cache έχει τα δικά της bit κατάστασης και κατά συνέπεια την δική της παράσταση στο διάγραμμα κατάστασης. Στο Σχήμα 18.8α φαίνονται οι μεταβάσεις που πραγματοποιούνται εξαιτίας των δράσεων τις οποίες ξεκινά ο επεξεργαστής που είναι προσκολλημένος σ' αυτή την μνήμη cache. Στο Σχήμα 18.8b φαίνονται οι μεταβάσεις που πραγματοποιούνται εξαιτίας γεγονότων που "κρυφακούο-



επιστρέφει ένα σήμα ότι διαμοιράζεται αυτή την γραμμή. Υστερα ο επεξεργαστής που ανταποκρίνεται *αλλάζει* την κατάσταση του αντιγράφου του από αποκλειστική σε διαμοιραζόμενη, και ο πρώτος επεξεργαστής διαβάζει την γραμμή από την κύρια μνήμη και *αλλάζει* την γραμμή στην μνήμη του cache από άκυρη σε διαμοιραζόμενη.

- Αν μια ή περισσότερες μνήμες cache διαθέτουν καθαρό αντίγραφο της γραμμής σε διαμοιραζόμενη κατάσταση, όπως σηματοδοτούν ότι διαμοιράζονται την γραμμή. Ο αρχικός επεξεργαστής διαβάζει την γραμμή και *αλλάζει* την κατάσταση της γραμμής στην μνήμη του cache από άκυρη σε διαμοιραζόμενη.
- Αν κάποια άλλη μνήμη cache διαθέτει τροποποιημένο αντίγραφο της γραμμής, τότε αυτή παρεμποδίζει την ανάγνωση της μνήμης και διαθέτει την γραμμή στην αιτούσα μνήμη cache από την διαμοιραζόμενη αρτηρία. Υστερα η μνήμη cache που ανταποκρίνεται *αλλάζει* την γραμμή της από τροποποιημένη σε διαμοιραζόμενη.<sup>3</sup>
- Αν καμία άλλη μνήμη cache δεν έχει αντίγραφο της γραμμής (καθαρής ή τροποποιημένης), τότε δεν επιστρέφονται σήματα. Ο αρχικός επεξεργαστής διαβάζει την γραμμή και την *αλλάζει* την γραμμή που βρίσκεται στην μνήμη του cache από άκυρη σε αποκλειστική.

### Προσπάθεια Ανάγνωσης

Όταν συμβεί προσπάθεια ανάγνωσης σε γραμμή που την στιγμή εκείνη βρίσκεται στην τοπική μνήμη cache, αλλά ο επεξεργαστής διαβάζει το αντικείμενο που χρειάζεται. Δεν υπάρχει *αλλαγή* κατάστασης: Η κατάσταση παραμένει τροποποιημένη, διαμοιραζόμενη, ή αποκλειστική.

### Αστοχία Εγγραφής

Όταν στην τοπική μνήμη cache συμβεί αστοχία εγγραφής, ο επεξεργαστής ξεκινά ανάγνωση μνήμης για να διαβάσει την γραμμή της κύριας μνήμης που περιέχει την διύθυνση που λείπει. Για τον σκοπό αυτό, ο επεξεργαστής εκδίδει ένα σήμα στην αρτηρία το οποίο σημαίνει *ανάγνωση με σκοπό την τροποποίηση* (read-with-intent-to-modify, RWITM). Όταν η γραμμή φορτωθεί, αμέσως σημειώνεται σαν τροποποιημένη. Σε σύγκριση με άλλες μνήμες cache, δύο δυνατά σενάρια προηγούνται της φόρτωσης της γραμμής δεδομένων.

Πρώτο, ίσως κάποια άλλη μνήμη cache να έχει ένα τροποποιημένο αντίγραφο αυτής της γραμμής (κατάσταση = τροποποιημένη). Στην περίπτωση αυτή, ο επεξεργαστής που έχει ειδοποιηθεί σηματοδοτεί τον αρχικό επεξεργαστή ότι ένας *άλλος* επεξεργα-



<sup>3</sup> Σε ορισμένες υλοποιήσεις, η μνήμη cache με την τροποποιημένη γραμμή σηματοδοτεί τον αρχικό επεξεργαστή να προσπαθήσει πάλι. Εν τω μεταξύ, ο επεξεργαστής με το τροποποιημένο αντίγραφο καταλαμβάνει την αρτηρία, εγγράφει την τροποποιημένη γραμμή πίσω στην κύρια μνήμη, και αλλάζει την γραμμή στην μνήμη του cache από τροποποιημένη σε διαμοιραζόμενη. Στη συνέχεια, ο αιτών επεξεργαστής προσπαθεί και πάλι και βρίσκει ότι ένας ή περισσότεροι επεξεργαστές έχουν ένα καθαρό αντίγραφο της γραμμής σε διαμοιραζόμενη κατάσταση, όπως περιγράφεται στο προηγούμενο σημείο.

στής έχει τροποποιημένο αντίγραφο της γραμμής. Ο αρχικός επεξεργαστής αφήνει την αρτηρία και περιμένει. Ο άλλος επεξεργαστής αποκτά προσπέλαση στην αρτηρία, εγγράφει την τροποποιημένη γραμμή μνήμης cache πίσω στην κύρια μνήμη, και αλλάζει την κατάσταση της γραμμής μνήμης cache σε άκυρη (επειδή ο αρχικός επεξεργαστής πρόκειται να τροποποιήσει την γραμμή αυτή). Στη συνέχεια, ο αρχικός επεξεργαστής θα εκδώσει και πάλι ένα σήμα RWITM προς την αρτηρία και ύστερα θα διαβάσει την γραμμή από την κύρια μνήμη, θα τροποποιήσει την γραμμή στην μνήμη cache, και θα σημειώσει την γραμμή με ένδειξη τροποποιημένης κατάστασης.

Το δεύτερο σενάριο είναι ότι καμία άλλη μνήμη cache δεν έχει τροποποιημένο αντίγραφο της αιτούμενης γραμμής. Στην περίπτωση αυτή, δεν επιστρέφεται σήμα, και ο αρχικός επεξεργαστής προχωράει σε ανίχνευση της γραμμής και τροποποίησή της. Εν τω μεταξύ, αν μία ή περισσότερες μνήμες cache διαθέτουν καθαρό αντίγραφο της γραμμής που βρίσκεται στην διαμοιραζόμενη κατάσταση, ακυρώνουν η κάθε μια το αντίγραφο τους της γραμμής, και αν κάποια μνήμη cache διαθέτει καθαρό αντίγραφο της γραμμής σε αποκλειστική κατάσταση, ακυρώνει το αντίγραφο της της γραμμής.

### Προσπάθεια Εγγραφής

Όταν συμβεί προσπάθεια εγγραφής σε γραμμή που την στιγμή εκείνη βρίσκεται στην τοπική μνήμη cache, το αποτέλεσμα θα εξαρτηθεί από την τρέχουσα κατάσταση της γραμμής στην τοπική μνήμη cache:

- ♦ **Διαμοιραζόμενη:** Πριν από την εκτέλεση της ανανέωσης ενημέρωσης, ο επεξεργαστής θα πρέπει να αποκτήσει αποκλειστική κυριότητα της γραμμής. Ο επεξεργαστής σηματοδοτεί την πρόθεσή του στην αρτηρία. Κάθε επεξεργαστής που διαθέτει διαμοιραζόμενο αντίγραφο της γραμμής στην μνήμη του cache αλλάζει τον τομέα από διαμοιραζόμενο σε άκυρο. Ύστερα ο αρχικός επεξεργαστής εκτελεί την ανανέωση ενημέρωσης και αλλάζει το αντίγραφο του από διαμοιραζόμενο σε τροποποιημένο.
- ♦ **Αποκλειστική:** Ο επεξεργαστής ήδη έχει αποκλειστικό έλεγχο αυτής της γραμμής, και έτσι αλλά εκτελεί την ανανέωση ενημέρωσης και αλλάζει το αντίγραφο του της γραμμής από αποκλειστικό σε τροποποιημένο.
- ♦ **Τροποποιημένη:** Ο επεξεργαστής ήδη έχει αποκλειστικό έλεγχο αυτής της γραμμής και έχει σημειωμένη την γραμμή σαν τροποποιημένη, και έτσι αλλά εκτελεί την ανανέωση ενημέρωσης.

### Συμφωνία Μνημών L1-L2

Μέχρι το σημείο αυτό περιγράψαμε πρωτόκολλα συμφωνίας μνημών cache με την βοήθεια της δραστηριότητας συνεργασίας μεταξύ μνημών cache που είναι συνδεδεμένες στην ίδια αρτηρία ή σε κάποια άλλη εγκατάσταση διασύνδεσης SMP. Συνήθως, αυτές οι μνήμες cache είναι μνήμες cache L2, και κάθε επεξεργαστής έχει, επίσης, μια μνήμη cache L1 που δεν συνδέεται απευθείας με την αρτηρία και που, κατά συνέπεια, δεν μπορεί να εμπλακεί σε κρυφό πρωτόκολλο. Έτσι, χρειάζεται κάποιος τρόπος διατήρησης της ακεραιότητας των δεδομένων και στα δύο επίπεδα και σε όλες τις μνήμες cache της διαμόρφωσης SMP.

Η στρατηγική είναι να επακτείνουμε το πρωτόκολλο MESI (ή οποιοδήποτε πρωτόκολλο συμφωνίας μνημών cache) στις μνήμες cache L1. Έτσι, κάθε γραμμή στην μνήμη cache L1 θα περιέχει bit ένδειξης της κατάστασης. Στην ουσία, αντικειμενικός σκοπός είναι ο παρακάτω: Σε κάθε γραμμή που βρίσκεται και στην μνήμη cache L2 και στην αντίστοιχη της μνήμη cache L1, η κατάσταση της γραμμής L1 θα πρέπει να παρακολουθεί την κατάσταση της γραμμής L2. Ένας απλός τρόπος για να γίνει αυτό είναι να υιοθετηθεί η τακτική εγγραφής προς τα μέσα στην μνήμη cache L1. Στην περίπτωση αυτή η εγγραφή προς τα μέσα θα γίνεται στην μνήμη cache L1 και όχι στην κύρια μνήμη. Η τακτική εγγραφής προς τα μέσα στην L1 προωθεί οποιαδήποτε τροποποίηση σε γραμμή L1 προς τα έξω στην μνήμη cache L2 και κατά συνέπεια την κάνει ορατή στις άλλες μνήμες cache L2. Η χρήση της τακτικής εγγραφής προς τα μέσα στην L1 ζητά το περιεχόμενο της L1 να είναι υποσύνολο του περιεχομένου της L2. Το γεγονός αυτό με την σειρά του υποδηλώνει ότι η προσεταιριστικότητα της μνήμης cache L2 θα πρέπει να είναι ίση με ή μεγαλύτερη από την προσεταιριστικότητα της L1. Η τακτική της εγγραφής προς τα μέσα στην L1 χρησιμοποιείται στον SMP S 390 της IBM.

Αν η μνήμη cache L1 έχει τακτική εγγραφής προς τα πίσω, η σχέση μεταξύ των δύο μνημών cache θα είναι πολύπλοκη. Υπάρχουν πολλές προσεγγίσεις για διατήρηση της συμφωνίας. Για παράδειγμα, η προσέγγιση που χρησιμοποιείται στον Pentium II περιγράφεται με λεπτομέρειες στο [SHAN98].

Ένας από τους εντονότερους νέους τομείς στην σχεδίαση συστημάτων υπολογιστών είναι η δημιουργία σημών. Η δημιουργία σημών είναι μια άλλη λύση για συμμετρική πολυεπεξεργασία σαν προσέγγιση που προσφέρει υψηλή απόδοση και μεγάλη διαθεσιμότητα και είναι ιδιαίτερα ελκυστική για εφαρμογές εξυπηρετητών (server). Μπορούμε να ορίσουμε το σημόν σαν μια ομάδα διασυνδεδεμένων ακέραιων υπολογιστών που λειτουργούν μαζί σαν ενωποιημένος πόρος υπολογισμών ο οποίος μπορεί να δημιουργήσει την ψευδαίσθηση ότι είναι μια μηχανή. Ο όρος *ακέραιος υπολογιστής* (whole computer) σημαίνει ένα σύστημα που μπορεί να λειτουργεί μόνο του, εκτός του σημόνου. Στην βιβλιογραφία, κάθε υπολογιστής σε σημόν συνήθως ονομάζεται *κόμβος* (node).

Ο [BREW97] αναφέρει τέσσερα ωφέλη που έχουμε με τα σημήν. Μπορούμε να τα θεωρήσουμε και σαν αντικειμενικούς σκοπούς ή σαν απαιτήσεις σχεδίασης:

- **Απόλυτη δυνατότητα κλιμάκωσης (absolute scalability):** Είναι δυνατή η δημιουργία μεγάλων σημών που ξεπερνούν κατά πολύ την ισχύ ακόμη και των μεγαλύτερων μηχανών. Ένα σημόν μπορεί να έχει δεκάδες μηχανών, από τις οποίες η κάθε μια είναι ένας πολυεπεξεργαστής.
- **Δυνατότητα αυξητικής κλιμάκωσης (incremental scalability):** Ένα σημόν είναι διαμορφωμένο με τέτοιο τρόπο ώστε να είναι δυνατή η πρόσθεση νέων συστημάτων στο σημόν κατά μικρές αυξήσεις. Έτσι, ένας χρήστης μπορεί να ξεκινήσει με ένα μέτριο σύστημα και να επεκτείνεται καθώς οι ανάγκες αυξάνουν, χωρίς να χρειάζεται μεγάλη αναβάθμιση στην οποία ένα μικρό υπάρχον σύστημα θα αντικα-