

Πανεπιστήμιο Θεσσαλίας
Τμήμα Πληροφορικής

Αρχιτεκτονική Υπολογιστών

Λυμένες Ασκήσεις

Ασκηση 1:

Έστω ένας επεξεργαστής, στον οποίο ένα πρόγραμμα ολοκληρώνει την εκτέλεσή του σε χρόνο T_1 . Σε μια νέα έκδοση του επεξεργαστή οι λειτουργίες συγκεκριμένου τύπου έχουν επιταχυνθεί επί ρ . Τι ποσοστό κύκλων μηχανής από την εκτέλεση του προγράμματος στον πρώτο επεξεργαστή αντιστοιχεί στις λειτουργίες που έχουν επιταχυνθεί, αν το πρόγραμμα ολοκληρώνει την εκτέλεσή του στο νέο επεξεργαστή σε χρόνο T_2 ; Σε τι ποσοστό κύκλων στο δεύτερο επεξεργαστή αντιστοιχεί το ποσοστό αυτό; Υποθέστε ότι ο δεύτερος επεξεργαστής λειτουργεί στην ίδια συχνότητα με τον πρώτο.

Απάντηση

Η βελτίωση της απόδοσης του επεξεργαστή σε λειτουργίες συγκεκριμένου τύπου μπορεί να γίνει είτε με υποστήριξη νέων εντολών που καθεμιά αντικαθιστά πολλαπλές εντολές του πρώτου επεξεργαστή, είτε με μείωση του αριθμού κύκλων μηχανής στην εκτέλεση κάποιων πολύπλοκων εντολών για τις λειτουργίες αυτές.

Στην πρώτη περίπτωση η βελτίωση οδηγεί σε μείωση του αριθμού εντολών που εκτελούνται στο ίδιο πρόγραμμα, ενώ στη δεύτερη περίπτωση η βελτίωση οδηγεί σε μείωση του μέσου αριθμού κύκλων μηχανής ανά εντολή (CPI).

Στη γενική περίπτωση μπορούμε να έχουμε βελτίωση και με τους δύο πιο πάνω τρόπους. Έτσι, έστω N_{A1} ο αριθμός των εντολών του προγράμματος στον πρώτο επεξεργαστή που αντιστοιχούν στις λειτουργίες που μας ενδιαφέρουν, N_{A2} ο αριθμός των εντολών αυτών στο δεύτερο επεξεργαστή, και N_B ο αριθμός των υπόλοιπων εντολών του προγράμματος, που είναι ο ίδιος και για τους δύο επεξεργαστές. Αντίστοιχα, έστω CPI_{A1} ο μέσος αριθμός κύκλων μηχανής ανά εντολή για τις N_{A1} παραπάνω εντολές του πρώτου επεξεργαστή, CPI_{A2} ο μέσος αριθμός κύκλων μηχανής ανά εντολή για τις N_{A2} παραπάνω εντολές του δεύτερου επεξεργαστή, και CPI_B ο μέσος αριθμός κύκλων μηχανής ανά εντολή για τις υπόλοιπες εντολές, που είναι ο ίδιος και για τους δύο επεξεργαστές.

Δεδομένου ότι οι δύο επεξεργαστές λειτουργούν στην ίδια συχνότητα, για την επιτάχυνση των βελτιωμένων λειτουργιών θα έχουμε:

$$\rho = N_{A1}CPI_{A1} / N_{A2}CPI_{A2}$$

ενώ η επιτάχυνση της εκτέλεσης του προγράμματος θα είναι:

$$T_1/T_2 = (N_{A1}CPI_{A1} + N_B CPI_B) / (N_{A2}CPI_{A2} + N_B CPI_B)$$

Γενικά $T_1/T_2 < \rho$, όταν $N_B CPI_B > 0$, και μάλιστα η επιτάχυνση του προγράμματος μειώνεται, όσο μειώνεται ο αριθμός κύκλων που καταναλώνεται στις βελτιωμένες λειτουργίες σε σχέση με τις υπόλοιπες. Το γεγονός αυτό επαληθεύει το νόμο του Amdahl, που μας λέει ότι η επιτάχυνση στο χρόνο εκτέλεσης του προγράμματος περιορίζεται από το μέρος εκείνου, του οποίου η εκτέλεση δεν έχει βελτιωθεί.

Αν το ζητούμενο ποσοστό κύκλων στον πρώτο επεξεργαστή είναι α_1 , τότε:

$$\alpha_1 = N_{A1}CPI_{A1} / (N_{A1}CPI_{A1} + N_B CPI_B)$$

οπότε από τις προηγούμενες σχέσεις παίρνουμε:

$$T_1/T_2 = \frac{1}{(N_{A2}CPI_{A2} + N_B CPI_B) / (N_{A1}CPI_{A1} + N_B CPI_B)} =$$

$$\begin{aligned}
&= \frac{1}{(N_{A1}CPI_{A1}/\rho + N_B CPI_B) / (N_{A1}CPI_{A1} + N_B CPI_B)} = \\
&= \frac{1}{\alpha_1 / \rho + (1 - \alpha_1)} \Rightarrow \\
\Rightarrow \alpha_1 (1 - 1/\rho) &= 1 - T_2/T_1 \Rightarrow \alpha_1 = (1 - T_2/T_1) / (1 - 1/\rho)
\end{aligned}$$

Αν το αντίστοιχο ποσοστό κύκλων μηχανής στο δεύτερο επεξεργαστή είναι α_2 , τότε:

$$\alpha_2 = N_{A2}CPI_{A2} / (N_{A2}CPI_{A2} + N_B CPI_B)$$

οπότε με όμοιο τρόπο:

$$T_1/T_2 = (\rho\alpha_2 + (1 - \alpha_2)) \Rightarrow \alpha_2 (\rho - 1) = T_1/T_2 - 1 \Rightarrow \alpha_2 = (T_1/T_2 - 1) / (\rho - 1)$$

Άσκηση 2:

Θεωρήστε την κλασική αρχιτεκτονική MIPS με μερική επικάλυψη. Έστω ότι θέλουμε να επεκτείνουμε την αρχιτεκτονική αυτή για υποστήριξη εντολών ALM με ένα τελούμενο από τη μνήμη, στις οποίες η τελική διεύθυνση προσπέλασης λαμβάνεται χωρίς μετατόπιση από κάποιον καταχωρητή γενικού σκοπού. Συγκεκριμένα, η εντολή:

`add $rd, $rt, ($rs)`

προσθέτει το περιεχόμενο του καταχωρητή \$rt με το περιεχόμενο της θέσης μνήμης με διεύθυνση που δίνεται από τον καταχωρητή \$rs, και αποθηκεύει το αποτέλεσμα στον καταχωρητή \$rd. Οι υπόλοιπες εντολές, όπως και οι εντολές ALM χωρίς προσπέλαση μνήμης, παραμένουν όπως ήταν.

A. Προσδιορίστε τις απαραίτητες φάσεις του κύκλου εντολής που να υλοποιούν το μηχανισμό επικάλυψης στη νέα αρχιτεκτονική.

B. Περιγράψτε τις γραμμές παροχέτευσης που απαιτούνται στην υλοποίηση της νέας αρχιτεκτονικής, ώστε να αντιμετωπίζονται επιτυχώς κίνδυνοι εξαρτήσεων από δεδομένα. Δώστε την απάντησή σας σε έναν πίνακα, ο οποίος να δίνει για κάθε συνδυασμό φάσεων που μπορούν να εμφανίσουν τέτοιους κινδύνους, δύο εντολές που να βρίσκονται σε αυτές.

Γ. Αναφέρετε τους κινδύνους εξαρτήσεων από δεδομένα που δεν αντιμετωπίζονται επιτυχώς με παροχέτευση, και δώστε παραδείγματα διαδοχής εντολών, όπου αυτοί εμφανίζονται.

Απάντηση

A. Για να προσδιορίσουμε τις απαιτούμενες φάσεις του κύκλου εντολής, θα ξεκινήσουμε με τις γνωστές μας φάσεις IF, ID, EX, MEM και WB της κλασικής αρχιτεκτονικής MIPS, και θα εξετάσουμε τι αλλαγές χρειαζόμαστε, τόσο για πιθανή προσθήκη νέων φάσεων, όσο και για αναδιάταξη τις σειρές τους στον κύκλο εντολής.

Όσο αφορά τη σειρά διαδοχής των παραπάνω φάσεων στον κύκλο εντολής, αυτή δε θα αλλάξει, εφ' όσον επιδιώκουμε την υποστήριξη στη νέα ΜΕΔ όλων των παλαιών εντολών. Πιο συγκεκριμένα, η φάση εκτέλεσης EX είναι απαραίτητο να προηγείται της φάσης προσπέλασης μνήμης MEM, ώστε η πρώτη να υπολογίζει την τελική διεύθυνση προσπέλασης για τη δεύτερη. Η σειρά διαδοχής των υπόλοιπων φάσεων είναι προφανής.

Επειδή θέλουμε οι εντολές ALM να μπορούν να διαβάσουν ένα τελούμενο από τη μνήμη, θα πρέπει να εισάγουμε μια φάση προσπέλασης μνήμης πριν από τη φάση EX, ή ισοδύναμα να εισάγουμε μια φάση εκτέλεσης πράξεων ALM μετά τη φάση MEM. Θα προτιμήσουμε τη δεύτερη λύση, μια που οι δύο φάσεις προσπέλασης μνήμης δημιουργούν κινδύνους από εξαρτήσεις στη μνήμη δεδομένων που είναι δύσκολο να ελεγχθούν, και θα εισάγουμε τη νέα φάση, έστω ALU, μεταξύ των φάσεων MEM και WB. Η τελική διαδοχή των φάσεων του κύκλου εντολής θα είναι επομένως:

IF, ID, EX, MEM, ALU, WB

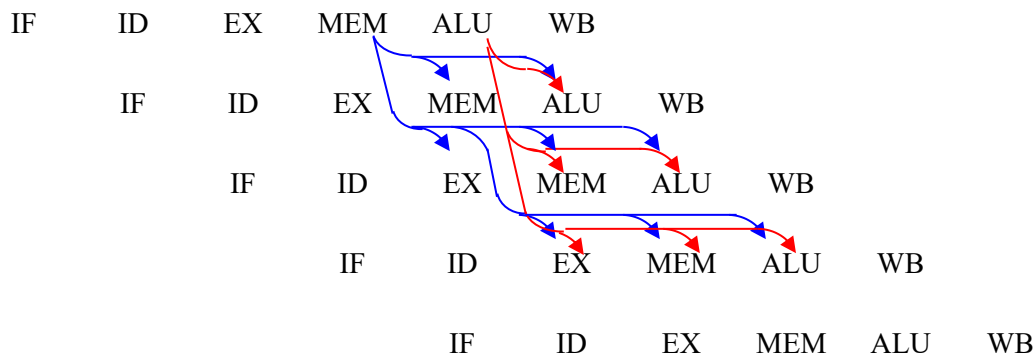
Παρατηρούμε ότι όλες οι εντολές ALM, παλιές και νέες, μπορούν να εκτελούνται στη φάση ALU, εφ' όσον σε καμία περίπτωση δεν έχουμε προσπέλαση μνήμης μετά την εκτέλεσή τους.

Έτσι, η φάση EX γίνεται μια φάση υπολογισμού τελικής διεύθυνσης προσπέλασης για τις εντολές φόρτωσης/αποθήκευσης και το υλικό της εκφυλίζεται σε έναν απλό αθροιστή.

Β. Κατ' αρχήν, επειδή η φάση EX – μετά την πιο πάνω μετατροπή – δεν παράγει αποτέλεσμα που να πρέπει να μπορεί να χρησιμοποιηθεί από επόμενες εντολές, δεν έχουμε παροχέτευση από τη φάση αυτή. Είναι αναγκαίο όμως να έχουμε παροχέτευση προς τη φάση EX, όπως θα δούμε στα παραδείγματα που ακολουθούν. Οι φάσεις MEM και ALU πρέπει να διαθέτουν γραμμές παροχέτευσης τόσο στην είσοδο, όσο και στην έξοδό τους, όπως και στην κλασική αρχιτεκτονική MIPS.

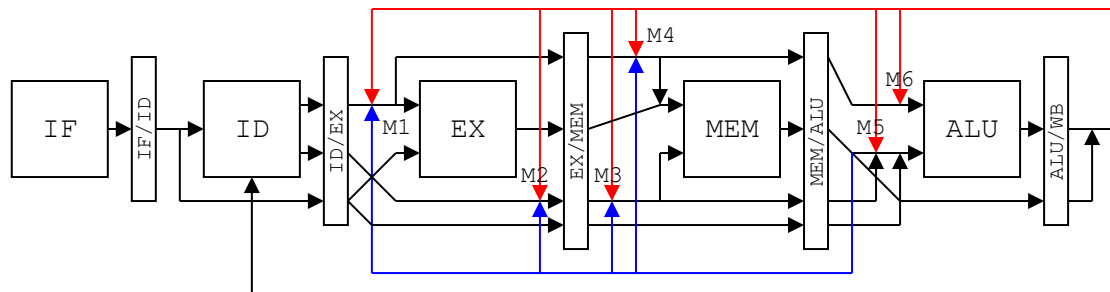
Το μέγιστο βάθος στο οποίο πρέπει να φτάνει η παροχέτευση είναι τρεις εντολές, επειδή τόσες είναι οι φάσεις που παρεμβάλλονται μεταξύ των ID και WB, με την υπόθεση ότι η ανάγνωση του ΦΚ γίνεται στον ίδιο κύκλο αμέσως μετά την εγγραφή του, κι επειδή η τελευταία από τις φάσεις αυτές πρέπει να μπορεί να παροχετεύσει δεδομένα προς την πρώτη. Αυτό σημαίνει ακόμα ότι η φάση ALU πρέπει να μπορεί να κρατήσει το αποτέλεσμά της στη ΜΕΔ για τρεις κύκλους μετά την παραγωγή του.

Όλα τα παραπάνω φαίνονται σχηματικά στο ακόλουθο διάγραμμα χρονισμού ιδανικής επικάλυψης 5 διαδοχικών εντολών:



Οι μπλε και κόκκινες γραμμές δίνουν τις απαιτούμενες παροχετεύσεις από την πρώτη από τις 5 εντολές προς τις επόμενες 3. Η τελευταία εντολή διαβάζει το αποτέλεσμα της πρώτης από το ΦΚ.

Η ΜΕΔ που υλοποιεί τον απαιτούμενο μηχανισμό παροχέτευσης φαίνεται στο ακόλουθο διάγραμμα:



Οι γραμμές παροχέτευσης ξεκινούν είτε από τον καταχωρητή επικάλυψης MEM/ALU (με μπλε) είτε από τον καταχωρητή επικάλυψης ALU/WB (με κόκκινο). Διακρίνονται 6 πολυπλέκτες παροχέτευσης για την επιλογή τιμών που προωθούνται, δύο για κάθε φάση που δέχεται παροχέτευση. Πρέπει να παρατηρήσουμε ότι πολλές φορές, και λόγω της πιθανά μεγάλης χρονικής απόστασης της παροχέτευσης, μπορεί να ενεργοποιηθεί κάποιος πολυπλέκτης που βρίσκεται πριν τη φάση προς την οποία απευθύνεται η παροχέτευση, και τα δεδομένα που προωθούνται να περνάνε μέσα από ενδιάμεσους καταχωρητές επικάλυψης. Έτσι, το μονοπάτι παροχέτευσης που ενεργοποιείται πιθανά να είναι αρκετά πολύπλοκο, γι' αυτό και απαιτείται προσεκτικός έλεγχος για τους 6 πολυπλέκτες, για κάθε περίπτωση παροχέτευσης.

Από τα παραπάνω σχηματίζουμε τον ακόλουθο πίνακα παραδειγμάτων για κάθε μια περίπτωση παροχέτευσης από κάποια εντολή E_i προς τις επόμενες τρεις, όπου διακρίνεται και το αντίστοιχο μονοπάτι παροχέτευσης που ενεργοποιείται:

Παροχέτευση	Παράδειγμα	Μονοπάτι παροχέτευσης
MEM(i) → MEM(i+1)	lw \$5,0(\$6) add \$7,\$1,(\$5)	MEM/ALU → M4 → MEM
MEM(i) → ALU(i+1)	lw \$5,0(\$6) add \$8,\$5,\$4	MEM/ALU → ALU/WB → M6 → ALU
MEM(i) → EX(i+2)	lw \$5,0(\$6) ... lw \$4,16(\$5)	MEM/ALU → M1 → EX
MEM(i) → MEM(i+2)	lw \$5,0(\$6) ... sw \$5,-4(\$7)	MEM/ALU → ALU/WB → M3 → MEM
MEM(i) → ALU(i+2)	lw \$5,0(\$6) ... addiu \$7,\$5,1	MEM/ALU → ALU/WB → M4 → MEM/ALU → ALU
MEM(i) → EX(i+3)	lw \$5,0(\$6) sw \$8,4(\$5)	MEM/ALU → ALU/WB → M1 → EX
MEM(i) → MEM(i+3)	lw \$5,0(\$6) or \$10,\$10,(\$5)	MEM/ALU → ALU/WB → M1 → EX/MEM → MEM
MEM(i) → ALU(i+3)	lw \$5,0(\$6) sub \$8,\$1,\$5	MEM/ALU → ALU/WB → M2 → EX/MEM → MEM/ALU → ALU
ALU(i) → ALU(i+1)	addiu \$7,\$4,-8 add \$8,\$7,(\$3)	ALU/WB → M5 → ALU
ALU(i) → MEM(i+2)	addiu \$7,\$4,-8 ... sw \$7,8(\$10)	ALU/WB → M3 → MEM
ALU(i) → ALU(i+2)	addiu \$7,\$4,-8 ... and \$6,\$6,\$7	ALU/WB → M3 → MEM/ALU → ALU
ALU(i) → EX(i+3)	addiu \$7,\$4,-8 lw \$6,0(\$7)	ALU/WB → M1 → EX
ALU(i) → MEM(i+3)	addiu \$7,\$4,-8 sw \$7,108(\$10)	ALU/WB → M2 → EX/MEM → MEM
ALU(i) → ALU(i+3)	addiu \$7,\$4,-8 ori \$5,\$7,2	ALU/WB → M1 → EX/MEM → MEM/ALU → ALU

Γ. Από το διάγραμμα χρονισμού φαίνεται ότι η παροχέτευση δεν αντιμετωπίζει επιτυχώς τους κινδύνους εξαρτήσεων από δεδομένα, όταν εμπλέκονται η φάση MEM μιας εντολής και η φάση EX της επόμενης, ή η φάση ALU μιας εντολής και μία από τις φάσεις EX και MEM της επόμενης, ή τη φάση EX της μεθεπόμενης εντολής.

Όπως και προηγουμένως, σχηματίζουμε τον ακόλουθο πίνακα παραδειγμάτων για τις περιπτώσεις αυτές:

Κίνδυνος	Παράδειγμα	Μονοπάτι παροχέτευσης
MEM(i) → EX(i+1)	lw \$5, 0 (\$6) lw \$8, 4 (\$5)	MEM/ALU → M1 → EX
ALU(i) → EX(i+1)	addiu \$7, \$4, -8 sw \$10, 0 (\$7)	ALU/WB → M1 → EX
ALU(i) → MEM(i+1)	addiu \$7, \$4, -8 sub \$9, \$5, (\$7)	ALU/WB → M4 → MEM
ALU(i) → EX(i+2)	addiu \$7, \$4, -8 ... lw \$10, 16 (\$7)	ALU/WB → M1 → EX

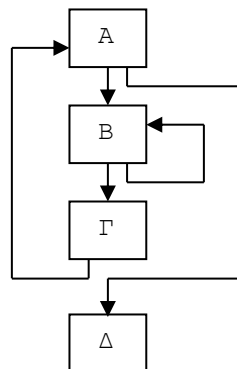
Όλες οι πιο πάνω περιπτώσεις κινδύνων αντιμετωπίζονται με πάγωμα του μηχανισμού επικάλυψης για έναν ή και δύο – για τη δεύτερη περίπτωση – κύκλους μηχανής, και στη συνέχεια με παροχέτευση μέσω του μονοπατιού που υποδεικνύεται.

Άσκηση 3:

Έστω μονάδα κινητής υποδιαστολής που εκτελεί κάποια σύνθετη πράξη με μερική επικάλυψη. Η μονάδα αποτελείται από 4 υπομονάδες, οι οποίες υλοποιούν την πράξη σε 5 μερικά επικαλυπτόμενες φάσεις, ως εξής:

- Η υπομονάδα Α υλοποιεί την πρώτη και την τέταρτη φάση της πράξης, που απαιτούν από έναν κύκλο μηχανής.
- Η υπομονάδα Β υλοποιεί τη δεύτερη φάση, ενώ η υπομονάδα Γ υλοποιεί την τρίτη φάση. Οι δύο αυτές φάσεις απαιτούν από 2 κύκλους μηχανής, με τον τελευταίο της πρώτης να επικαλύπτεται με τον πρώτο της δεύτερης.
- Τέλος, η υπομονάδα Δ υλοποιεί την πέμπτη και τελευταία φάση, η οποία απαιτεί έναν κύκλο μηχανής.

Η εκτέλεση της πράξης στη μονάδα μπορεί να αναπαρασταθεί με το ακόλουθο σχηματικό διάγραμμα:



- Βρείτε τον πίνακα δέσμευσης και το διάγραμμα σύγκρουσης της μονάδας.
- Κατασκευάστε το ελαχιστοποιημένο διάγραμμα καταστάσεων και βρείτε τον κύκλο μέγιστου ρυθμού ολοκλήρωσης πράξεων.
- Υπολογίστε ένα άνω φράγμα στο ρυθμό ολοκλήρωσης πράξεων, με βάση τον αριθμό κύκλων μηχανής που καταλαμβάνει κάθε φάση.
- Αν ο κύκλος που βρήκατε στο ερώτημα Β δεν παρουσιάζει το ρυθμό ολοκλήρωσης πράξεων που υπολογίσατε στο ερώτημα Γ, τροποποιήστε τη μονάδα, εισάγοντας σε κατάλληλα σημεία υπομονάδες καθυστέρησης που να αλλάζουν το σχετικό χρονισμό των υπομονάδων, ώστε να αυξήσετε το ρυθμό ολοκλήρωσης πράξεων στο μέγιστο δυνατό.

Απάντηση

Για ιδανική επικάλυψη μεταξύ διαδοχικών πράξεων, μια μονάδα κινητής υποδιαστολής πρέπει να αποφεύγει την επαναχρησιμοποίηση υπομονάδων, ή τη χρησιμοποίηση υπομονάδων για χρόνο μεγαλύτερο από έναν κύκλο μηχανής, ώστε οι διαδοχικές πράξεις να μην εμφανίζουν δομικές εξαρτήσεις μεταξύ τους. Κάτι τέτοιο πολλές φορές δεν επιτυγχάνεται, κυρίως επειδή λόγω κόστους κάποιες υπομονάδες δεν ακολουθούν τους πιο πάνω περιορισμούς. Έτσι, η επικάλυψη δεν είναι ιδανική, και ο σχεδιασμός της μονάδας πρέπει να γίνεται με σκοπό τη μεγιστοποίηση του ρυθμού ολοκλήρωσης πράξεων.

A. Από το σχηματικό διάγραμμα της μονάδας και την περιγραφή του χρονισμού των φάσεων που μας δίνεται λαμβάνουμε τον παρακάτω πίνακα δέσμευσης. Η υπομονάδα A καταλαμβάνει τον 1ο και τον 5ο κύκλο μηχανής, η υπομονάδα B τον 2ο και τον 3ο, η υπομονάδα Γ τον 3ο και τον 4ο, ενώ η υπομονάδα Δ καταλαμβάνει τον 6ο κύκλο μηχανής μιας πράξης:

	1	2	3	4	5	6
A	X				X	
B		X	X			
Γ			X	X		
Δ						X

Το διάνυσμα σύγκρουσης της μονάδας προκύπτει από τον πίνακα δέσμευσης δοκιμάζοντας να ξεκινήσουμε μια νέα πράξη για καθέναν από τους κύκλους 2-6 της εκτέλεσης της προηγούμενης. Ξεκινώντας μια πράξη στον 2ο κύκλο, προκύπτει σύγκρουση στον αμέσως επόμενο κύκλο στην υπομονάδα B. Νέες πράξεις μπορούν να ξεκινήσουν στον 3ο και στον 4ο κύκλο, ενώ δε μπορούν να ξεκινήσουν στον 5ο κύκλο, λόγω της επαναχρησιμοποίησης της υπομονάδας A. Τέλος, μια νέα πράξη μπορεί να ξεκινήσει στον 6ο κύκλο χωρίς σύγκρουση. Το διάνυσμα σύγκρουσης θα είναι επομένως το:

10010

και υποδηλώνει τις συγκρούσεις στον 2ο και στον 5ο κύκλο της προηγούμενης πράξης (ή ισοδύναμα στον 1ο και 4ο κύκλο μετά το ξεκίνημα της προηγούμενης πράξης).

B. Για να βρούμε το ζητούμενο διάγραμμα καταστάσεων ξεκινάμε με αρχική κατάσταση αυτήν που περιγράφεται από το διάνυσμα σύγκρουσης 10010 και αντιστοιχεί στην εκτέλεση μιας πράξης στη μονάδα μας.

Δύο κύκλους μηχανής αργότερα μπορούμε να ξεκινήσουμε μια νέα πράξη, γεγονός που μας οδηγεί σε μια νέα κατάσταση που περιγράφεται από το διάνυσμα που προκύπτει από τη λογική διάζευξη του διανύσματος της προηγούμενης, ολισθημένου κατά 2 θέσεις αριστερά, με το διάνυσμα σύγκρουσης:

$01000 \text{ OR } 10010 = 11010$

Το παραπάνω διάνυσμα μας δείχνει ότι μια τρίτη πράξη μπορεί να ξεκινήσει με καθυστέρηση 3 ή 5 κύκλων από το ξεκίνημα της δεύτερης.

Αν η δεύτερη πράξη ξεκινήσει 3 κύκλους μετά την πρώτη, λαμβάνουμε την κατάσταση που περιγράφεται από διάνυσμα που προκύπτει από τη λογική διάζευξη του διανύσματος της αρχικής κατάστασης, ολισθημένου κατά 3 θέσεις αριστερά, με το διάνυσμα σύγκρουσης:

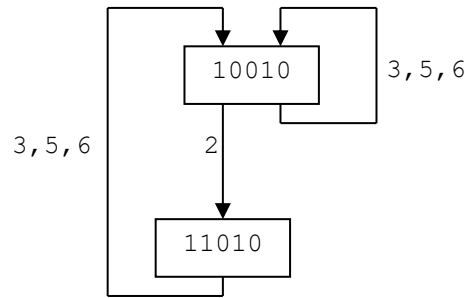
$10000 \text{ OR } 10010 = 10010$

Το διάνυσμα αυτό είναι το ίδιο με το διάνυσμα σύγκρουσης, κι επομένως η κατάσταση που λάβαμε ταυτίζεται με την αρχική.

Παρόμοια οδηγούμαστε στην αρχική κατάσταση αν η δεύτερη πράξη ξεκινήσει 5 κύκλους μετά την πρώτη, ή αργότερα.

Από τη δεύτερη πιο πάνω κατάσταση, με καθυστέρηση 3, 5, ή περισσότερων κύκλων λαμβάνουμε πάλι την αρχική κατάσταση.

Έτσι, το ελαχιστοποιημένο διάγραμμα καταστάσεων θα είναι το εξής:



όπου οι ενδείξεις κύκλων μηχανής υποδηλώνουν τον κύκλο μετά το ξεκίνημα μιας πράξης που μπορεί να ξεκινήσει η επόμενη. Στον 6ο κύκλο ή αργότερα από το ξεκίνημα μιας πράξης μπορούμε πάντα να αρχίσουμε μια νέα πράξη και να οδηγηθούμε στην αρχική κατάσταση.

Το παραπάνω διάγραμμα περιέχει δύο κύκλους, από τους οποίους ο πρώτος περιλαμβάνει μόνο την αρχική κατάσταση και αντιστοιχεί σε ρυθμούς ολοκλήρωσης μιας πράξης κάθε 3, 5, 6 ή περισσότερους κύκλους μηχανής. Προφανώς ο μέγιστος ρυθμός ολοκλήρωσης πράξεων για τον κύκλο αυτό είναι μια πράξη κάθε 3 κύκλους μηχανής. Ο δεύτερος κύκλος του διαγράμματος καταστάσεων περιλαμβάνει και τις δύο καταστάσεις, και δίνει ρυθμούς ολοκλήρωσης δύο πράξεων (αφού κάθε κατάσταση αντιστοιχεί στο ξεκίνημα μιας πράξης) κάθε 5, 7, 8 ή περισσότερους κύκλους μηχανής. Ο μέγιστος ρυθμός ολοκλήρωσης πράξεων για τον κύκλο αυτό είναι 2 πράξεις κάθε 5 κύκλους μηχανής, που είναι και ο βέλτιστος για όλο το διάγραμμα, αφού $2/5 > 1/3$.

Γ. Στον πίνακα δέσμευσης παρατηρούμε ότι ο μέγιστος αριθμός κατειλημμένων κύκλων μηχανής σε κάποια από τις φάσεις της εκτέλεσης μιας πράξης είναι 2 κύκλοι. Αυτό σημαίνει ότι σε χρόνο 6 κύκλων μηχανής που διαρκεί μια πράξη μπορούν να εκτελεστούν το πολύ $6/2 = 3$ πράξεις. Κάτι τέτοιο θα είναι εφικτό, αν το επιτρέψουν οι συγκρούσεις στο χρονισμό διαδοχικών πράξεων. Αν οι συγκρούσεις είναι τέτοιες, ώστε να μην επιτρέπουν την εκτέλεση κάποιας φάσης σε 1 ή περισσότερους από τους 6 κύκλους μηχανής που διαρκεί μια πράξη, τότε στο χρόνο των 6 κύκλων μηχανής θα εκτελούνται λιγότερες από 3 πράξεις.

Επομένως, ο αριθμός των 3 πράξεων σε 6 κύκλους μηχανής μας δίνει ένα άνω φράγμα στο ρυθμό ολοκλήρωσης διαδοχικών πράξεων ίσο με $3/6 = 1/2$.

Δ. Ο βέλτιστος ρυθμός ολοκλήρωσης που βρήκαμε από τον αρχικό πίνακα δέσμευσης λαμβάνοντας υπ' όψη τις συγκρούσεις στο χρονισμό των διαδοχικών πράξεων είναι μικρότερος από το παραπάνω άνω φράγμα. Μπορούμε όμως να τροποποιήσουμε τον πίνακα δέσμευσης – και κατ' επέκταση και τη μονάδα που σχεδιάζουμε – εισάγοντας κατάλληλες καθυστερήσεις μεταξύ διαδοχικών κύκλων μηχανής, ώστε να εξαλείψουμε τις συγκρούσεις που δε μας επιτρέπουν να λάβουμε ρυθμό ολοκλήρωσης πράξεων ίσο με το άνω φράγμα.

Για να επιτύχουμε λοιπόν ρυθμό ολοκλήρωσης διαδοχικών πράξεων $1/2$, θέλουμε να μπορούμε να ξεκινάμε μια νέα πράξη κάθε δεύτερο κύκλο μηχανής. Αυτό σημαίνει ότι αν η πρώτη φάση μιας πράξης εκτελείται στον κύκλο 1, τότε οι θέσεις των κύκλων 3,5,... της υπομονάδας A στον πίνακα δέσμευσης είναι απαγορευμένες για το υπόλοιπο της πράξης, επειδή δεσμεύονται για επόμενες πράξεις. Ξαναγράφοντας τον πίνακα δέσμευσης, ώστε να λαμβάνει υπ' όψη τις απαγορευμένες θέσεις, έχουμε μέχρι τώρα:

	1	2	3	4	5	6
A	X		F		F	
B						
Γ						
Δ						

Παρόμοια σκεφτόμενοι, βρίσκουμε ότι για τη δεύτερη φάση που υλοποιείται στην υπομονάδα B στους κύκλους 2 και 3, απαγορευμένες θέσεις είναι οι θέσεις όλων των υπόλοιπων κύκλων

4,5,6,7,..., το οποίο δε μας ενοχλεί, αφού δεν ακολουθεί επαναχρησιμοποίηση της B για την ίδια πράξη. Προσθέτοντας τις απαγορευμένες θέσεις της τρίτης φάσης και της υπομονάδας Γ, ο πίνακας δέσμευσης γίνεται:

	1	2	3	4	5	6
A	X		F		F	
B		X	X	F	F	F
Γ			X	X	F	F
Δ						

Στη συνέχεια πρέπει να προσθέσουμε στον πίνακα την τέταρτη φάση, η οποία υλοποιείται πάλι στην υπομονάδα A. Επειδή όμως ο κύκλος 5, στον οποίο αρχικά αυτή βρισκόταν, αντιστοιχεί σε απαγορευμένη θέση για την υπομονάδα A, πρέπει να βρούμε κάποιον επόμενο κύκλο μηχανής από τις μη απαγορευμένες θέσεις, έστω τον 6. Για να εκτελέσουμε όμως την τέταρτη φάση στον κύκλο αυτόν, θα πρέπει να εισάγουμε ένα στοιχείο καθυστέρησης, το οποίο να κρατήσει το αποτέλεσμα της προηγούμενης, μέχρι τον κύκλο 6. Έτσι, ο πίνακας δέσμευσης διαμορφώνεται ως εξής:

	1	2	3	4	5	6	7
A	X		F		FD	X	F
B		X	X	F	F	F	F
Γ			X	X	F	F	F
Δ							

Η πέμπτη φάση, τέλος, εκτελείται μετά την τέταρτη στον κύκλο 7, ο οποίος προέκυψε με την εισαγωγή του στοιχείου καθυστέρησης που επιμήκυνε το συνολικό χρόνο εκτέλεσης της πράξης κατά έναν κύκλο μηχανής. Ο τελικός πίνακας δέσμευσης θα είναι επομένως:

	1	2	3	4	5	6	7
A	X				FD	X	
B		X	X				
Γ			X	X			
Δ							X

Η καθυστέρηση που στον πίνακα σημειώνεται με D μπορεί να υλοποιηθεί εύκολα με έναν καταχωρητή.

Αν θέλουμε να επανασχεδιάσουμε το σχηματικό διάγραμμα της μονάδας μετά την πιο πάνω τροποποίηση, θα πάρουμε το ακόλουθο διάγραμμα:

