



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ
ΒΙΟΜΗΧΑΝΙΚΗΣ ΣΧΕΔΙΑΣΗΣ & ΠΑΡΑΓΩΓΗΣ

Μηχανές Πεπερασμένων Καταστάσεων

Όνομα Σπουδαστή: ΓΖΟΥΝΤΕΛΛΗΣ ΟΔΥΣΣΕΑΣ

Όνομα Επιβλέποντα Καθηγητή: ΔΡΟΣΟΣ ΧΡΗΣΤΟΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ

ΙΑΝΟΥΑΡΙΟΣ 2019

ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ
Π. Ράλλη & Θηβών 250, 12244 Αιγάλιω, Αθήνα – Ελλάδα

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος ΓΖΟΥΝΤΕΛΛΗΣ ΟΔΥΣΣΕΑΣ, του ΕΥΣΤΑΡΤΙΟΥ, φοιτητής του Τμήματος Αυτοματισμού του Πανεπιστημίου Δυτικής Αττικής, πριν αναλάβω την εκπόνηση της Πτυχιακής Εργασίας μου, δηλώνω ότι ενημερώθηκα για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε, ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα, σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασή της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση Π.Ε με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε πρέπει να ολοκληρώσει εντός τουλάχιστον ενός ημερολογιακού 6μήνου από την ημερομηνία ανάθεσής της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18. παρ.5 του ισχύοντος Εσωτερικού Κανονισμού».

Ο Δηλών

Ημερομηνία

Περιεχόμενα

ΠΡΟΛΟΓΟΣ.....	5
ΜΕΡΟΣ Α' : ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ	
<u>ΚΕΦΑΛΑΙΟ 1</u> ΜΗΧΑΝΕΣ ΠΕΠΕΡΑΣΜΕΝΩΝ ΚΑΤΑΣΤΑΣΕΩΝ	
1.1 Εισαγωγικές έννοιες	6
<u>ΚΕΦΑΛΑΙΟ 2</u> ΑΚΟΛΟΥΘΙΑΚΑ ΚΥΚΛΩΜΑΤΑ	
2.1 Εισαγωγή.....	6
2.2 Κατηγορίες ακολουθιακών κυκλωμάτων.....	7
2.3 Κατηγορίες φλιπ – φλοπ.....	8
2.4 Καταχωρητές.....	8
<u>ΚΕΦΑΛΑΙΟ 3</u> ΓΕΝΙΚΗ ΔΟΜΗ ΜΗΧΑΝΩΝ ΠΕΠΕΡΑΣΜΕΝΩΝ ΚΑΤΑΣΤΑΣΕΩΝ	
3.1 Ορισμός.....	9
3.2 Μνήμη ROM.....	10
<u>ΚΕΦΑΛΑΙΟ 4</u> ΜΕΘΟΔΟΙ ΑΝΑΠΑΡΑΣΤΑΣΗΣ	
4.1 Διάγραμμα μετάβασης καταστάσεων.....	11
4.2 Πίνακας μετάβασης καταστάσεων.....	12
4.3 Διάγραμμα ροής.....	14
4.4 Σύγκριση.....	15
<u>ΚΕΦΑΛΑΙΟ 5</u> ΤΥΠΟΙ ΜΗΧΑΝΩΝ ΠΕΠΕΡΑΣΜΕΝΩΝ ΚΑΤΑΣΤΑΣΕΩΝ	
5.1 Εισαγωγή Moore, Mealy.....	16
5.2 Ασύγχρονη Moore.....	17
5.3 Σύγχρονη Moore.....	17
5.4 Ασύγχρονη Mealy.....	17
5.5 Σύγχρονη Mealy.....	18
5.6 Σύγκριση Moore, Mealy.....	18
<u>ΚΕΦΑΛΑΙΟ 6</u> ΑΛΓΟΡΙΘΜΟΣ ΣΧΕΔΙΑΣΗΣ ΜΠΚ	
6.1 Βασικά βήματα σχεδίασης.....	21
<u>ΚΕΦΑΛΑΙΟ 7</u> ΕΦΑΡΜΟΓΕΣ	

7.1 Μηχανή πώλησης.....	22
7.2 Ελεγκτής φωτός κυκλοφορίας.....	23
ΚΕΦΑΛΑΙΟ 8 ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ	
8.1 Αναπαράσταση ενός πρωτοκόλλου.....	24
8.2 Επαλήθευση της ορθότητας του ABP.....	26
8.3 Μεταφορά του διαγράμματος σε ηλεκτρονικό υπολογιστή....	27
ΜΕΡΟΣ Β' : ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ	
ΚΕΦΑΛΑΙΟ 9 ΔΙΑΚΟΠΤΗΣ ΝΥΧΤΟΣ	
9.1 Εισαγωγή.....	30
ΚΕΦΑΛΑΙΟ 10 ΔΙΑΔΙΚΑΣΙΑ ΣΧΕΔΙΑΣΗΣ	
10.1 Κατανόηση των προδιαγραφών.....	30
10.2 Μέθοδοι αναπαράστασης	31
10.3 Ελαχιστοποίηση Καταστάσεων.....	37
10.4 Δυαδική κωδικοποίηση.....	38
10.5 Επιλογή τύπου φλιπ – φλοπ.....	39
10.6 Τελικό ακολουθιακό κύκλωμα.....	40
ΚΕΦΑΛΑΙΟ 11 ΒΧ-24	
11.1 Εισαγωγικές έννοιες.....	41
11.2 Κύκλωμα εφαρμογής.....	43
11.3 Δομή προγράμματος.....	44
ΕΠΙΛΟΓΟΣ.....	47
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	48
ΠΑΡΑΡΤΗΜΑ Α ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ.....	49
ΠΑΡΑΡΤΗΜΑ Β ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ.....	50
ΠΑΡΑΡΤΗΜΑ Γ DATASHEETS.....	51

ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΕΦΑΡΜΟΓΗ ΤΩΝ ΜΗΧΑΝΩΝ **ΠΕΠΕΡΑΣΜΕΝΩΝ ΚΑΤΑΣΤΑΣΕΩΝ**

ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ

ΠΡΟΛΟΓΟΣ

Οι περισσότεροι από εμάς σκεφτόμαστε τους υπολογιστές ως lap-top, PCs. Ωστόσο, υπάρχουν πολλές περισσότερες μικρές συσκευές υπολογισμού σε λειτουργία, όλες γύρω από μας. Κάθε μία από αυτές τις συσκευές μπορεί να θεωρηθεί ως ένα *αντιδραστικό σύστημα* και αυτό επειδή κάθε μία από αυτές λειτουργεί με την αντίδραση στα σήματα ή στις εισόδους από τον εξωτερικό κόσμο. Η βασική παρατήρηση που μπορεί να γίνει για τα αντιδραστικά συστήματα είναι ότι η αντίδραση σ'ένα ειδικό ερέθισμα δεν είναι το ίδιο σε κάθε περίπτωση. Για παράδειγμα, η περίπτωση μιας μηχανής εισιτηρίων χώρων

στάθμευσης, δεν θα τυπώσει ένα εισιτήριο όταν πιέζουμε το κουμπί, εκτός αν έχουμε παρεμβάλλει ήδη κάποια χρήματα.

Το επίσημο πρότυπο μιας Μηχανής Πεπερασμένων Καταστάσεων (ΜΠΚ) αναπτύχθηκε στις αρχές της δεκαετίας του '50, για την μελέτη των προβλημάτων στη σχεδίαση των συνδυαστικών και διαδοχικών κυκλωμάτων.

Συγκεκριμένα, η ιδέα της “κατάστασης” ως βασική έννοια στην αναπαράσταση ενός συστήματος εισήχθη αρχικά το 1936, από τον A.M.Turing (στους υπολογιστικούς αριθμούς). Αργότερα, η έννοια υιοθετήθηκε από τον C.E.Shannon για την θεωρία πληροφοριών (‘Μαθηματική θεωρία στις επικοινωνίες’). Στη συνέχεια, η ιδέα της “κατάστασης”, επανεισήχθη από τους D.A.Huffman (‘Η σύνθεση των διαδοχικών δικτύων-switching’), S.C.Kleene (‘Αναπαράσταση των γεγονότων σε νευρωτικά δίκτυα και πεπερασμένα αυτόματα’) και E.F.Moore (Πειράματα στις διαδοχικές μηχανές, των ‘Μελετών Αυτομάτων’), όπου και έγινε αποδεκτό ως μια από τις θεμελιώδεις έννοιες στην θεωρία των αυτόματων συστημάτων.

Το πρότυπο Μηχανών Πεπερασμένων Καταστάσεων (ΜΠΚ) περιορίζει τον αριθμό διαφορετικών αποκρίσεων σ’ένα ειδικό ερέθισμα, για να είναι πεπερασμένο και για να καθοριστεί από την περιγραφή της μηχανής. Αυτή είναι και η μεγάλη διαφορά μεταξύ του μοντέλου ΜΠΚ και των άλλων υπολογιστικών μοντέλων. Βασικά, στο πρότυπο ΜΠΚ έχουμε τη δυνατότητα κατασκευής μηχανών, όπου το χαρακτηριστικό τους γνώρισμα θα είναι ο πεπερασμένος αριθμός διαφορετικών καταστάσεων που θα έχουν.

Η ΜΠΚ είναι ένα εργαλείο που διαμορφώνει την επιθυμητή συμπεριφορά ενός διαδοχικού(ακολουθιακού) συστήματος. Ο σχεδιαστής πρέπει να αναπτύξει ένα πρότυπο πεπερασμένων καταστάσεων της συμπεριφοράς του συστήματος και έπειτα να σχεδιάσει ένα κύκλωμα που εφαρμόζει αυτό το πρότυπο. Δηλαδή, είναι μια τεχνική που επιτρέπει τον απλό και ακριβή σχεδιασμό των διαδοχικών λειτουργιών λογικής και ελέγχου. Εάν σχεδιάζουμε προγράμματα υπολογιστών, διαδοχικά κυκλώματα λογικής ή ηλεκτρονικά συστήματα ελέγχου, θα είμαστε σε θέση να καταστήσουμε το σχέδιο μας από την αρχική περίπλοκη μορφή σε μια πιο απλή και κατανοητή.

Υπάρχουν σχεδόν τόσες πολλές παραλλαγές του βασικού προτύπου μιας ΜΠΚ όσες είναι και οι εφαρμογές. Έτσι, εξετάζοντας μερικά απλά συστήματα που αντιμετωπίζουμε κάθε μέρα στη ζωή μας, μπορούμε να διακρίνουμε μια πόρτα ή μια ηλεκτρική λάμπα. Και στις δύο περιπτώσεις μπορούμε να προσδιορίσουμε δύο καταστάσεις, η πόρτα ανοιχτή ή κλειστή και η ηλεκτρική λάμπα με ανοιχτό η κλειστό φως.

Most of us think of computers as lap-top, PCs. However, there are many more small computing devices in operation, all around us. Each of these devices can be considered as a reactive system, because each of them works by reacting to signals or inputs from the outside world. The basic observation that can be made for reactive systems is that the reaction to a particular stimulus is not the same in every case. For example, the case of a parking ticket machine will not print a ticket when we press the button unless we already have some money.

The formal model of a Finite-Stage Engineer (MKK) was developed in the early 1950s to study the problems of composite and sequential circuit design.

In particular, the concept of "state" as a basic concept in the representation of a system was originally introduced in 1936, by A. M. Turing (in the computations). Later, the concept was adopted by C.E.Shannon for Information Theory ('Mathematical Theory in Communications'). Then, the idea of " state " was re-introduced by DA Huffman ('Switching'), SCKleene ('Representation of Events in Neurotic Networks and Finite Automatic') and EFMoore sequential machines, of "Automated Studies"), where it was accepted as one of the fundamental concepts in the theory of automated systems.

The Finite-Stage Machine (CCM) model limits the number of different responses to a particular stimulus, to be finite and to be determined by the machine description. This is the big difference between the MCA model and other computational models. Basically, in the MCA model, we have the ability to construct machines where their characteristic feature will be the finite number of different situations they will have.

The MPC is a tool that shapes the desired behavior of a sequential (sequential) system. The designer must develop a finite state model of the behavior of the system and then design a circuit that applies this pattern. That is, it is a technique that allows the simple and accurate design of sequential logic and control functions. If we design computer programs, sequential logic circuits or electronic control systems, we will be able to make our project from the original complex form to a simpler and more comprehensible one.

There are almost as many variations of the basic model of an MCC as are the applications. So by looking at some simple systems we face every day in our lives, we can distinguish a door or an electric lamp. In both cases, we can identify two states, the door open or closed and the light bulb with open or closed light.

Κεφάλαιο 1 **Μηχανές Πεπερασμένων Καταστάσεων**

1.1 Εισαγωγικές έννοιες

Οι Μηχανές Πεπερασμένων Καταστάσεων (ΜΠΚ) είναι συσκευές που αποκρίνονται σ' ένα σύνολο ερεθισμάτων (γεγονότα) με την παραγωγή των προβλέψιμων απαντήσεων (ενέργειες), βασισμένων σε μια ιστορία των προγενέστερων γεγονότων (επικρατούσα κατάσταση). Η "ιστορία" της μηχανής συνοψίζεται στην τιμή των εσωτερικών καταστάσεων της. Δημιουργούμε ένα σύνολο από καταστάσεις και μεταβάσεις για να μοντελοποιήσουμε τις αλληλεπιδράσεις ανάμεσα στις εισόδους και εξόδους. Όταν μια νέα είσοδος παρουσιαστεί στην ΜΠΚ, παράγεται μια έξοδος, η οποία εξαρτάται από αυτήν την είσοδο και την παρούσα κατάσταση, και η μηχανή μεταβαίνει σε νέα(καινούργια) κατάσταση, καλούμενη ως επόμενη κατάσταση. Αυτή η νέα κατάσταση εξαρτάται τόσο από την είσοδο όσο και από την παρούσα κατάσταση. Η εσωτερική κατάσταση αποθηκεύεται σε ένα μπλοκ με την ονομασία "μνήμη". Περιγράφουμε μια ΜΠΚ ως εξής:

$$M = (S, I, O, Fs, Fo)$$

Ένα σύνολο πεπερασμένων καταστάσεων (S): Αντιπροσωπεύουν τις ιδιαίτερες διαμορφώσεις που η μηχανή μας μπορεί να υποθέσει. Κάθε κατάσταση δίνει μια πλήρη περιγραφή του συστήματος σε οποιαδήποτε στιγμή, ωστόσο το γεγονός ότι μας επιτρέπεται να έχουμε μόνο ένα σταθερό αριθμό καταστάσεων, περιορίζει τι μπορούμε να υπολογίσουμε (ή να διαμορφώσουμε) με μια ΜΠΚ.

Ένα σύνολο πεπερασμένων εισόδων και εξόδων (I,O): Εξετάζουμε τις αλληλεπιδράσεις των εισόδων και εξόδων, ενώ παράλληλα προσδιορίζουμε και τα γεγονότα, που καθορίζουν τις διάφορες εισόδους που μια μηχανή θα αναγνωρίσει.

Την κατάσταση έναρξης (So): Είναι η κατάσταση της μηχανής προτού να έχει λάβει οποιαδήποτε γεγονότα, θεωρείται ως η αρχική κατάσταση της ΜΠΚ. Ωστόσο, πολλές φορές μπορούμε και να την παραβλέψουμε.

Την λειτουργία μετάβασης των καταστάσεων (Fs): Η παρούσα κατάσταση της μηχανής σε συνδυασμό με την είσοδο θα μας δώσει την νέα κατάσταση.

Την λειτουργία της εξόδου (Fo): Η τρέχουσα κατάσταση της μηχανής σε συνδυασμό με την είσοδο θα μας δώσει την έξοδο. Η συγκεκριμένη λειτουργία καθορίζει και τον τύπο μιας ΜΠΚ.

Σύμφωνα με όσα αναφέραμε στη γενική περιγραφή για τις ΜΠΚ, παρατηρούμε ότι η λειτουργία τους βασίζεται στη δομή των ακολουθιακών κυκλωμάτων. Για αυτό το λόγο θεωρούμε σκόπιμο να περιγράψουμε τα βασικά χαρακτηριστικά των διαδοχικών κυκλωμάτων, ώστε να γίνει πιο ευκρινής και κατανοητή η δομή των ΜΠΚ.

Κεφάλαιο 2

ΑΚΟΛΟΥΘΙΑΚΑ ΚΥΚΛΩΜΑΤΑ

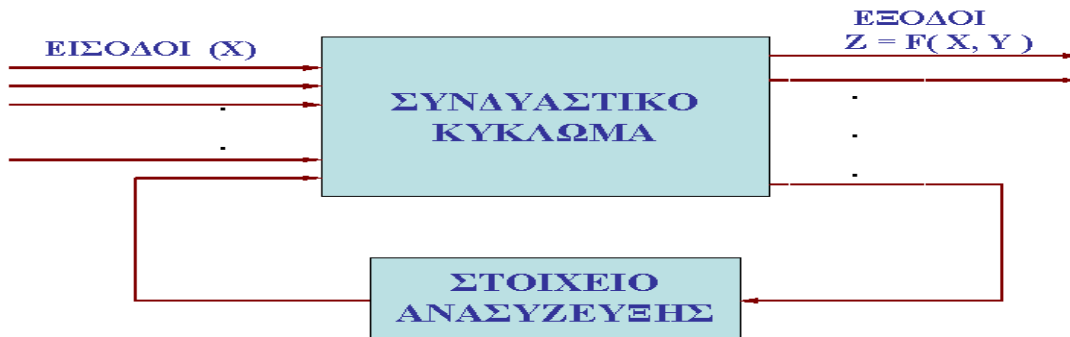
2.1 Εισαγωγή

Είναι τα συστήματα εκείνα τα οποία όταν διεγερθούν, εκτελούν μια σειρά (ακολουθία) διαδικασιών προς ολοκλήρωση μιας εργασίας, μετά δε την ολοκλήρωση αυτή επανέρχονται στην αρχική τους κατάσταση, ώστε να είναι έτοιμα για μια δεύτερη τέτοια ακολουθία, κ.ο.κ... Χαρακτηριστικό των συστημάτων αυτών είναι ότι “θυμούνται” την προηγούμενή τους κατάσταση, για να αποφασίσουν για την επόμενη, κατόπιν κατάλληλης εισόδου.

Ένα παράδειγμα μηχανικού ακολουθιακού συστήματος είναι η κλειδαριά των χρηματοκιβωτίων, η οποία για να ανοίξει, πρέπει να σχηματισθεί μια ακολουθία αριθμών, ώστε να σχηματισθεί ο κατάλληλος συνδυασμός. Η ακολουθία αυτή μπορεί να διακοπεί και να ολοκληρωθεί σε έναν επιθυμητό χρόνο, και αυτό γιατί το σύστημα “θυμάται” σε ποια κατάσταση σταμάτησε. Όταν συμπληρωθεί η ακολουθία, δίνεται το σήμα να ανοίξει η κλειδαριά και το σύστημα επανέρχεται στην αρχική κατάσταση για να είναι έτοιμο για άλλη παρόμοια ενέργεια.

Αυτά τα λογικά κυκλώματα (συστήματα), είναι κυκλώματα των οποίων η έξοδος δεν είναι συναρτήσει μόνο της παρούσας εισόδου, αλλά και της καταστάσεως που βρίσκεται το κύκλωμα πριν εφαρμοσθεί η είσοδος, δηλαδή το κύκλωμα αυτό έχει την ιδιότητα της “μνήμης”, ή αλλιώς την ικανότητα αποθήκευσης δυαδικών πληροφοριών. Τα κυκλώματα αυτά ονομάζονται ακολουθιακά κυκλώματα (Sequential Circuits ή Finite Automata), και αποτελούνται βασικά από ένα συνδυαστικό κύκλωμα με επί πλέον

χαρακτηριστικό την ύπαρξη ανασυζεύξεως (Feedback) μεταξύ εξόδου-εισόδου, μέσω ενός στοιχείου ανασυζεύξεως, όπως φαίνεται και στο σχήμα 2.1



Σχήμα 2.1 Ακολουθιακό Κύκλωμα ή Finite Automata

2.2 Κατηγορίες ακολουθιακών κυκλωμάτων

Υπάρχουν δύο κατηγορίες ακολουθιακών κυκλωμάτων :

- ⇒ **Τα σύγχρονα ακολουθιακά κυκλώματα**, όπου η εφαρμογή των εισόδων, η εκτέλεση των διαφόρων λειτουργιών, καθώς και η λήψη της εξόδου γίνεται σε καθορισμένο χρόνο, μέσω της χρησιμοποίησης ‘ρολογιού’ (Clock), ή γεννήτρια τετραγωνικών παλμών.
- ⇒ **Τα ασύγχρονα ακολουθιακά κυκλώματα**, στα οποία οι διάφορες λειτουργίες δεν επιτελούνται σε ένα καθορισμένο χρόνο (δεν υπάρχει “ρολόι”), ενώ η ταχύτητα εκτέλεσης τους εξαρτάται από αυτό το κύκλωμα και μόνο.

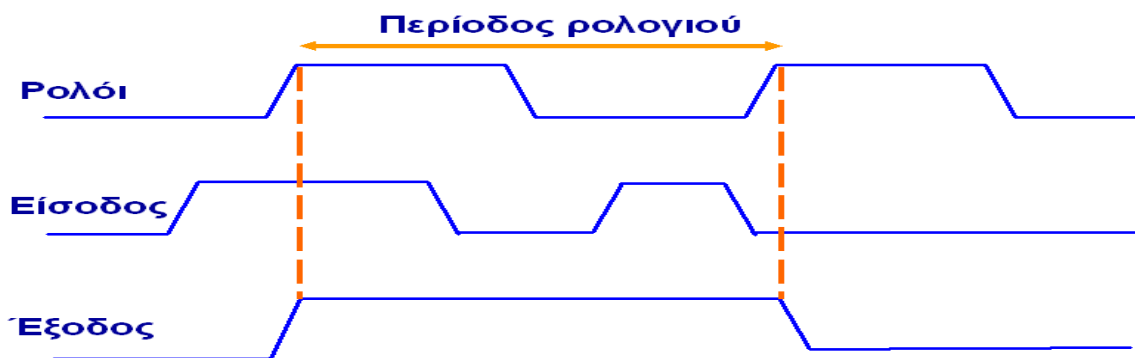
Επειδή, τα ακολουθιακά κυκλώματα χαρακτηρίζονται από την ιδιότητα της “μνήμης”, χρησιμοποιούν για την πραγματοποίησή τους εκτός των βασικών λογικών κυκλωμάτων (AND, OR, NAND, NOR κ.λ.π.) και ένα στοιχείο μνήμης το οποίο συνήθως είναι το δικατάστατο ηλεκτρονικό κύκλωμα που φέρεται υπό το όνομα Flip-Flop.

2.3 Κατηγορίες φλιπ – φλοπ

Ως βασικό στοιχείο στοιχείο μνήμης στα ακολουθιακά κυκλώματα χρησιμοποιείται συνήθως το Flip-Flop, το οποίο δύναται να αποθηκεύσει μια δυαδική πληροφορία του ενός bit, εφόσον τούτο λαμβάνει μια από τις δύο σταθερές καταστάσεις 0 ή 1, και παραμένει σε αυτήν μέχρι νέας διεγέρσεως.

Γενικώς, ένα Flip-Flop έχει μια ή περισσότερες εισόδους ελέγχου ή δεδομένων (Data Inputs), και μια είσοδο όπου εφαρμόζεται ο ωρολογιακός παλμός, καθώς επίσης και δύο εξόδους : την Q, η οποία είναι η κανονική έξοδος και την Q', η οποία είναι πάντα το συμπλήρωμα της Q.

Συνήθως, η αλλαγή της καταστάσεως του Flip-Flop δεν γίνεται με την εφαρμογή των δεδομένων στις αντίστοιχες εισόδους, αλλά με την άφιξη του ωρολογιακού παλμού. Έτσι επιτυγχάνεται η σύγχρονος λειτουργία (ταυτόχρονη αλλαγή) μέσω αυτού του παλμού, πολλών Flip-Flops, τα οποία ανήκουν σε ένα ακολουθιακό κύκλωμα.



Σχήμα 2.2 Χρονισμός φλιπ – φλοπ

Ακμοπυρίδοτα είναι τα φ.φ. στα οποία επιτρέπεται η αλλαγή των περιεχομένων τους, όταν παρουσιάζονται λογικές καταστάσεις στις εισόδους μόνο κατά την διάρκεια των ακμών του παλμού χρονισμού. Συγκεκριμένα, έχουμε τα ακμοπυρίδοτα φ.φ. θετικής κατεύθυνσης, όπου στα ολοκληρωμένα αυτά κυκλώματα η μεταφορά πληροφοριών από την είσοδο στην έξοδο λαμβάνει χώρα κατά τη διάρκεια της θετικής ακμής του παλμού. Στο σχήμα 2.2 διακρίνουμε ένα ακμοπυρίδοτο θετικής κατεύθυνσης D φ-φ.

Επίσης, έχουμε και τα ακμοπυρίδοτα φ.φ. αρνητικής κατεύθυνσης, όπου στα ολοκληρωμένα αυτά κυκλώματα η μεταφορά πληροφοριών από την είσοδο στην έξοδο λαμβάνει χώρα κατά τη διάρκεια της αρνητικής ακμής του παλμού. Τέλος, διακρίνουμε και τα **επιπεδοπυρίδοτα** φ.φ., που οι λογικές καταστάσεις εφαρμόζονται στις εισόδους του φ.φ. κατά τη διάρκεια που ο παλμός του ρολογιού έχει σταθερή αρνητική ή θετική λογική τιμή (δηλαδή για θετική λογική 0V και 5V αντίστοιχα).

2.4 Καταχωρητές

Για την αποθήκευση μιας πληροφορίας του ενός bit, μπορεί να χρησιμοποιηθεί το κύκλωμα του Flip-Flop. Όταν η πληροφορία περιλαμβάνει περισσότερα του ενός bits, τότε για την αποθήκευσή της, απαιτείται μια σειρά από Flip-Flops και τόσα όσα είναι τα bits της πληροφορίας. Το σύνολο αυτό των F-F καλείται καταχωρητής (register).

Ο απλούστερος καταχωρητής είναι ο στατικός ο οποίος αποτελείται από ανεξάρτητα F-F (πχ D), στα οποία δύναται να αποθηκευθεί (εγγραφεί) μια πληροφορία των n bits και να ληφθεί (διαβασθεί) αυτή, όταν χρειασθεί.

Κεφάλαιο 3

Γενική Δομή Μηχανών Πεπερασμένων Καταστάσεων

3.1 Ορισμός

Μια απλή μηχανή καταστάσεων μπορεί να αποτελείται από συνδυαστικές λογικές, καταχωρητές εξόδου και καταχωρητές καταστάσεων. Εκτός από τις συσκευές εισόδων και εξόδων, η μηχανή καταστάσεων αποτελείται από δύο βασικά στοιχεία : Την συνδυαστική λογική και την μνήμη, δηλαδή τους καταχωρητές. Η μνήμη χρησιμοποιείται για να αποθηκεύσει τις καταστάσεις της μηχανής. Η συνδυαστική λογική μπορεί να αντιμετωπισθεί ως δύο ξεχωριστά μπλοκ λειτουργιών (σχήμα 3.2). Τον αποκωδικοποιητή επόμενης κατάστασης και τον αποκωδικοποιητή εξόδου. Ο πρώτος

υπολογίζει την επόμενη κατάσταση της μηχανής, ενώ ο δεύτερος παράγει τις πραγματικές εξόδους. Παρόλο που εκτελούν δύο διαφορετικές λειτουργίες, αυτές συνήθως συνδυάζονται σε μια ενιαία συνδυαστική λογική. Το βασικό μπλοκ διάγραμμα μιας ΜΠΚ βλέπουμε και στο σχήμα 3.1.

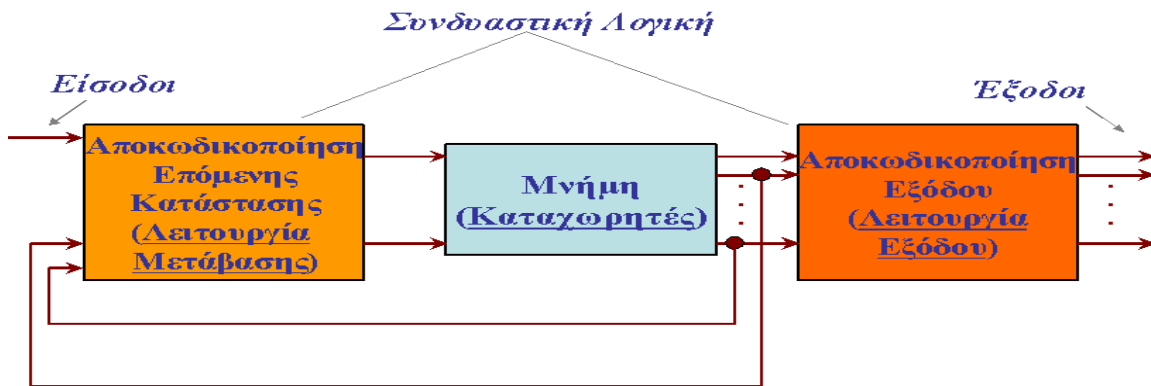


Σχήμα 3.1 Μπλοκ Διάγραμμα μιας απλής Μηχανής Πεπερασμένων Καταστάσεων

Η βασική λειτουργία μιας μηχανής καταστάσεων είναι διπλή :

Περιστρέφεται μέσω μιας ακολουθίας καταστάσεων, όπου η επόμενη κατάσταση υπολογίζεται από τον αποκωδικοποιητή επόμενης κατάστασης, ανάλογα με την παρούσα κατάσταση και τους όρους της εισόδου.

Παρέχει τις ακολουθίες των σημάτων εξόδου, που βασίζονται στις μεταβάσεις των καταστάσεων. Οι έξοδοι παράγονται από τον αποκωδικοποιητή εξόδου, ο οποίος βασίζεται στην παρούσα κατάσταση και τους όρους της εισόδου (αυτό δεν ισχύει πάντα). Σύνθετα ακολουθιακά κυκλώματα παρέχουν την ικανότητα της επανάληψης ακολουθιών (looping) και υπορουτίνων. Οι μεταβάσεις από μια κατάσταση σε άλλη καλείται έλεγχος αλληλουχίας και η λογική που απαιτείται για την απόφαση ή εκτέλεση των επόμενων καταστάσεων καλείται λειτουργία μετάβασης.

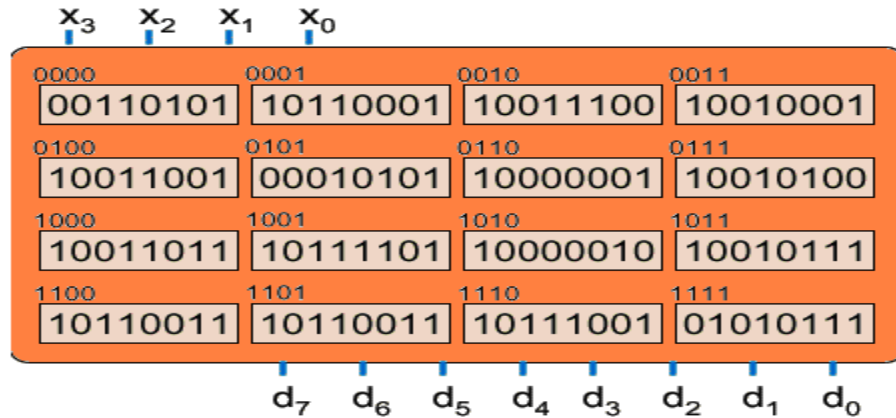


Σχήμα 3.2 Μηχανή Πεπερασμένων Καταστάσεων με ξεχωριστούς αποκωδικοποιητές

3.2 Μνήμη ROM

Σε μια Μηχανή Πεπερασμένων Καταστάσεων έχουμε τη δυνατότητα αντί των συνδυαστικών κυκλωμάτων, να τοποθετήσουμε μια μνήμη ROM. Όπως γνωρίζουμε η ROM (μνήμη αποκλειστικής ανάγνωσης) έχει ως κύριο χαρακτηριστικό τα δεδομένα που

εμπεριέχονται σε αυτήν να είναι αποθηκευμένα μόνιμα εκεί από όταν αυτή κατασκευάζεται. Βεβαίως, έχουμε τη δυνατότητα να χρησιμοποιήσουμε μνήμες ROM οι οποίες επαναπρογραμματίζονται (EPROM, EEPROM).



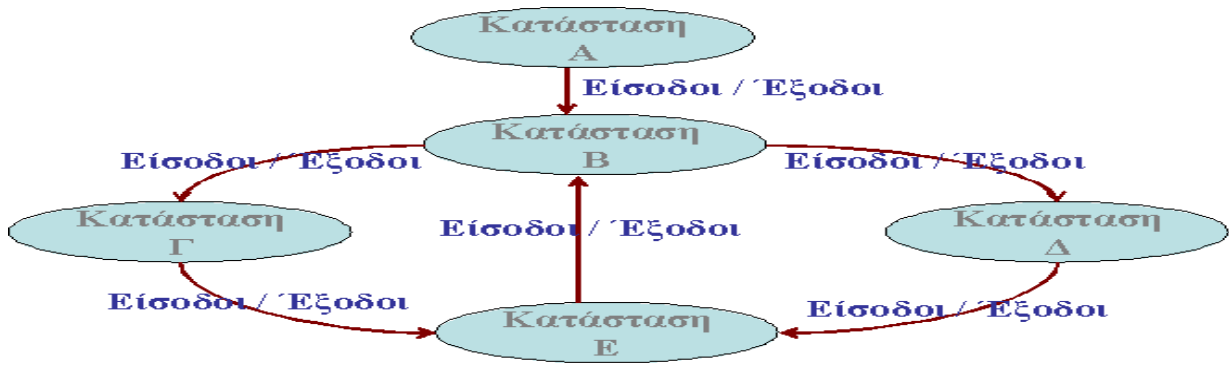
Σχήμα 3.3 Λειτουργία μνήμης ROM

Έτσι, σύμφωνα με το σχήμα 3.3 μπορούμε να εξηγήσουμε πως μια ROM μπορεί να χρησιμοποιηθεί αντί ενός συνδυαστικού κυκλώματος εισόδου σε μια ΜΠΚ. Στους καταχωρητές που εμπεριέχονται στην ROM θα είναι αποθηκευμένες οι καταστάσεις της μηχανής. Άρα, λαμβάνοντας τα σήματα εισόδου και τις επόμενες καταστάσεις οι γραμμές x αυτόματα καθορίζουν και τη διεύθυνση του καταχωρητή. Αυτός περιέχει τα δυφία εκείνα που περιγράφουν την κατάσταση που επιθυμούμε να εξάγουμε στις γραμμές d και που οδηγούμε στα αντίστοιχα Flip-Flop.

ΚΕΦΑΛΑΙΟ 4 ΜΕΘΟΔΟΙ ΑΝΑΠΑΡΑΣΤΑΣΗΣ

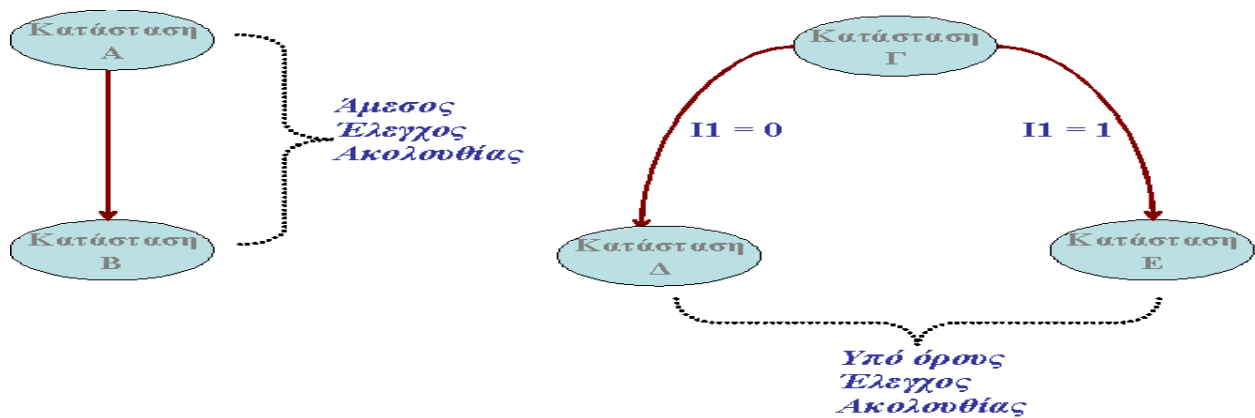
4.1 Διάγραμμα μετάβασης καταστάσεων

Η συμπεριφορά μιας Μηχανής Πεπερασμένων Καταστάσεων μπορεί να καθορισθεί σε γραφική μορφή όπως παρουσιάζεται στο σχήμα 4.1. Αυτό καλείται διάγραμμα καταστάσεων ή διάγραμμα μετάβασης καταστάσεων. Κάθε κύκλος αναπαριστά μια κατάσταση και κάθε βέλος αναπαριστά μια μετάβαση μεταξύ των καταστάσεων. Οι εισοδοί που προκαλούν τις μεταβάσεις δείχνονται πάνω σε κάθε βέλος μετάβασης.



Σχήμα 4.1 Διάγραμμα καταστάσεων 1

Ο έλεγχος ακολουθίας αναπαρίσταται στο διάγραμμα μετάβασης καταστάσεων όπως παρουσιάζεται στο σχήμα 4.2. Η άμεση ακολουθία ελέγχου απαιτεί μια άνευ όρων μετάβαση από την κατάσταση Α στην κατάσταση Β. Ομοίως, η υπό όρους ακολουθία ελέγχου παρουσιάζει μια υπό όρους μετάβαση από την κατάσταση Γ είτε στην κατάσταση Δ είτε στην κατάσταση Ε, ανάλογα το σήμα εισόδου Π.



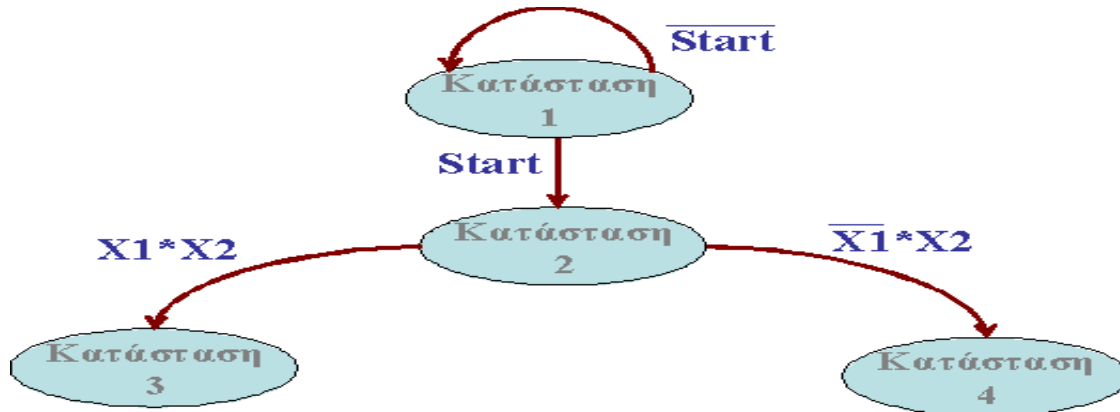
Σχήμα 4.2 Διάγραμμα καταστάσεων 2

Για τις μηχανές Moore η έξοδος που παράγεται αναπαριστάται στους κύκλους και ταυτίζεται με τις καταστάσεις. Ομοίως, στις μηχανές Mealy η υπό όρους έξοδος που παράγεται αναπαρίσταται στα βέλη μεταβάσεων μαζί με τις αντίστοιχες εισόδους. Παρουσιάζονται στο σχήμα. 4.3



Σχήμα 4.3 Διάγραμμα καταστάσεων για μηχανές Mo και Me

Τα διαγράμματα μετάβασης καταστάσεων μπορούν να γίνουν πιο συμπαγείς με το γράψιμο στις μεταβάσεις όχι των τιμών εισόδων που προκάλεσαν την μετάβαση, αλλά με μια Boolean έκφραση, καθορίζοντας τον συνδυασμό εισόδου ή τους συνδυασμούς που προκαλούν την μετάβαση. Για παράδειγμα, στο σχήμα 4.4 ορισμένες μεταβάσεις έχουν παρουσιαστεί για μια μηχανή με εισόδους Start, X1 και X2. Στη μετάβαση μεταξύ των καταστάσεων 1 και 2, οι εισόδοι X1 και X2 αγνοούνται (δηλαδή, είναι ‘αδιάφορες’) και έτσι δεν εμφανίζονται στο διάγραμμα. Αυτό κάνει και τη λειτουργία προφανέστερη.



Σχήμα 4.4 Διάγραμμα καταστάσεων 3

4.2 Πίνακας μετάβασης καταστάσεων

Μια δεύτερη μέθοδος για την αναπαράσταση ΜΠΚ είναι η πινακοειδής μορφή γνωστή ως πίνακας μεταβάσεων κατάστασης. Για κάθε κατάσταση ελέγχου της μηχανής ο πίνακας διευκρινίζει ένα σύνολο κανόνων μετάβασης. Υπάρχει ένας κανόνας ανά σειρά στον πίνακα, και συνήθως περισσότεροι από ένας κανόνες ανά κατάσταση. Ο πίνακας 1 του παραδείγματος περιέχει τους κανόνες μετάβασης για τον έλεγχο καταστάσεων, όπου τις έχουμε ονομάσει A και B. Κάθε κανόνας μετάβασης έχει τέσσερα μέρη, κάθε μέρος που αντιστοιχεί σε μια από τις τέσσερις στήλες στον πίνακα. Τα δύο πρώτα είναι όροι που πρέπει να ικανοποιηθούν για τον κανόνα μετάβασης για να είναι εκτελέσιμος. Διευκρινίζουν την παρούσα κατάσταση καθώς και την τιμή ενός σήματος εισόδου.

Όροι		Επίδραση	
Παρούσα Κατάσταση	Είσοδος	Επόμενη Κατάσταση	Έξοδος
A	-	A	1
A	1	B	0
B	0	B	0
B	1	A	1

Πίνακας 1 Καταστάσεων Mealy

Οι τελευταίες δύο στήλες του πίνακα καθορίζουν την επίδραση της εφαρμογής ενός κανόνα μετάβασης. Διευκρινίζουν πώς το “περιβάλλον” της μηχανής αλλάζει, π.χ., πώς η τιμή ενός σήματος εξόδου αλλάζει. Κατά συνέπεια, η νέα κατάσταση όπου η μηχανή μας φθάνει εάν εφαρμοστεί ο κανόνας μετάβασης.

Στο παραδοσιακό πρότυπο ΜΠΚ, το περιβάλλον της μηχανής αποτελείται από δύο πεπερασμένα και χωριστά σύνολα σημάτων: τα σήματα εισόδου και τα σήματα εξόδου. Κάθε σήμα έχει μια αυθαίρετη, αλλά πεπερασμένη, σειρά των πιθανών τιμών. Ο όρος που πρέπει να ικανοποιηθεί για τον κανόνα μετάβασης για να είναι εκτελέσιμος διατυπώνεται ως όρος στην τιμή από κάθε σήμα εισόδου, και η επίδραση της μετάβασης μπορεί να είναι μια αλλαγή των τιμών των σημάτων εξόδου. Η μηχανή στον πίνακα επεξηγεί εκείνο το πρότυπο. Έχει ένα σήμα εισόδου, και ένα σήμα εξόδου. Μια παύλα σε μια από τις πρώτες δύο στήλες χρησιμοποιείται σαν μια στενογραφία για να δείξει έναν “αδιάφορο” όρο. Ένας κανόνας μετάβασης, με μια παύλα στην πρώτη στήλη ισχύει για όλες τις καταστάσεις της μηχανής, και ένας κανόνας μετάβασης με μια παύλα στη δεύτερη στήλη ισχύει για όλες τις πιθανές τιμές του σήματος εισόδου. Οι παύλες στις τελευταίες δύο στήλες μπορούν να χρησιμοποιηθούν για να δείξουν ότι η εκτέλεση ενός κανόνα μετάβασης δεν αλλάζει το περιβάλλον. Μια παύλα στην τέταρτη στήλη σημαίνει ότι το σήμα εξόδου δεν αλλάζει, και ομοίως, μια παύλα στην τρίτη στήλη σημαίνει ότι η κατάσταση ελέγχου παραμένει αμετάβλητη.

Σε κάθε ειδική κατάσταση της μηχανής μπορούν να υπάρξουν μηδενικοί ή περισσότεροι κανόνες μετάβασης που είναι εκτελέσιμοι. Εάν κανένας κανόνας μετάβασης δεν είναι εκτελέσιμος, η μηχανή λέγεται ότι είναι σε end-state. Εάν ειδικά ένας κανόνας μετάβασης είναι εκτελέσιμος η μηχανή κάνει μια **deterministic** (υπολογιστική) κίνηση σε μία νέα κατάσταση ελέγχου. Εάν περισσότεροι από ένας κανόνες μετάβασης είναι εκτελέσιμοι, μια **nondeterministic** (στοχαστική) επιλογή γίνεται για να επιλέξει έναν κανόνα μετάβασης. Μια στοχαστική επιλογή σε αυτό το πλαίσιο σημαίνει ότι το κριτήριο επιλογής είναι απροσδιόριστο.

Όροι		Επίδραση	
Παρούσα Κατάσταση	Είσοδος	Επόμενη Κατάσταση	Έξοδος
A	-	B	0
A	0	Γ	0

Πίνακας 2 Καταστάσεων (Nondeterministic)

Ο πίνακας 2 επεξηγεί αυτή την έννοια. Δύο κανόνες μετάβασης καθορίζονται για την κατάσταση ελέγχου A. Εάν το σήμα εισόδου είναι ένα, μόνο ο πρώτος κανόνας είναι

εκτελέσιμος. Εάν το σήμα εισόδου είναι μηδέν, και οι δύο κανόνες θα είναι εκτελέσιμοι και η μηχανή θα κινηθεί είτε προς την κατάσταση B είτε προς την κατάσταση Γ.

Ο πρώτος πίνακας που παρουσιάσαμε προσδιορίζει την μηχανή Mealy. Στη μηχανή Moore διακρίνουμε δύο πίνακες. Ο ένας καθορίζει την επόμενη κατάσταση και ο άλλος αντιστοιχεί τις καταστάσεις της μηχανής με τις ανάλογες εξόδους. Έτσι, παρατηρούμε ότι η κατάσταση A αντιστοιχεί στην έξοδο 1 και η κατάσταση B αντιστοιχεί στην έξοδο 0 (πίνακας 3).

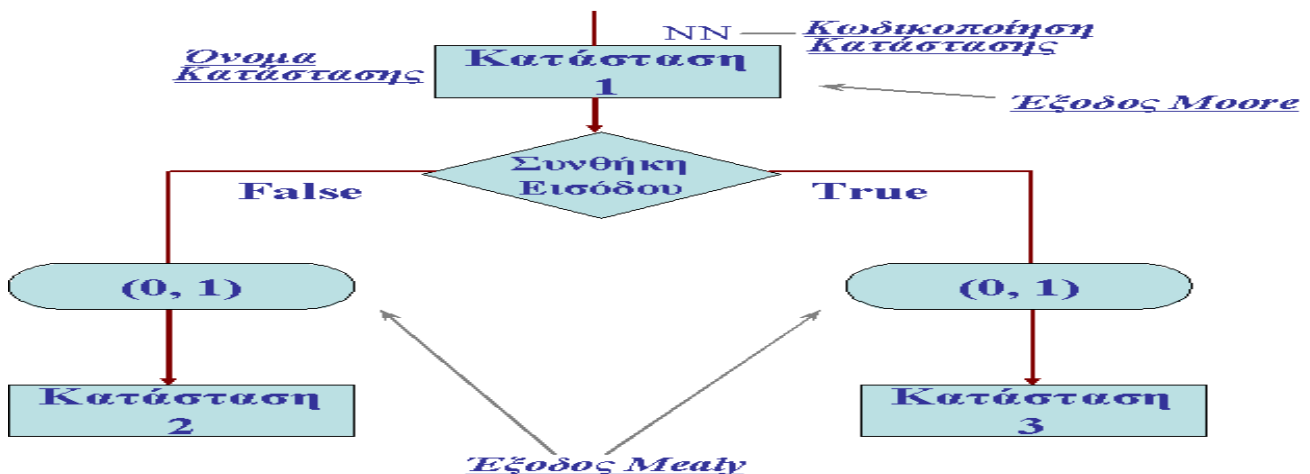
Παρούσα Κατάσταση	Είσοδος	Επόμενη Κατάσταση
A	1	A
A	0	B
B	1	B
B	0	A

Κατάσταση	Έξοδος
A	1
B	0

Πίνακας 3 Καταστάσεων Moore

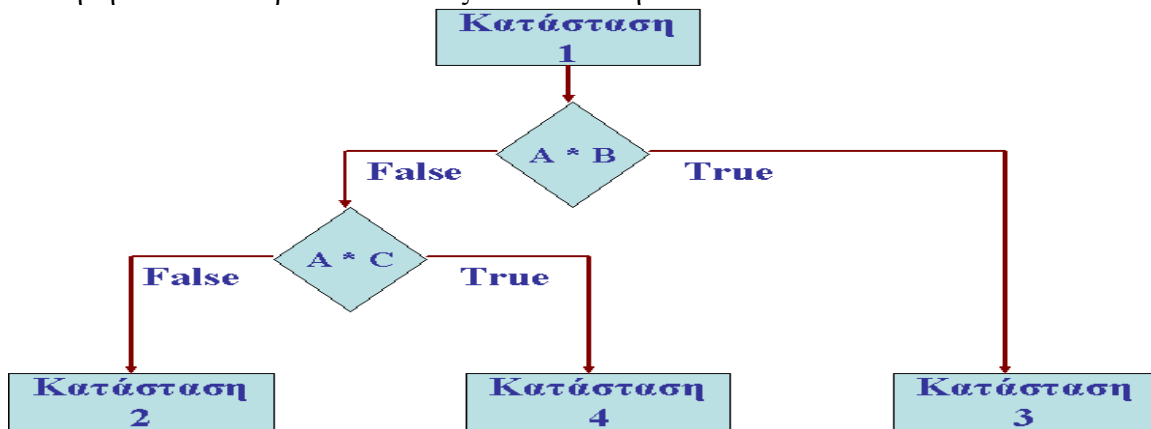
4.3 Διάγραμμα ροής

Μια τρίτη μέθοδος είναι βασισμένη στα διαγράμματα ροής. Εδώ, οι καταστάσεις αναπαρίστανται σε σχήμα ορθογώνιου. Τα στοιχεία μπορεί να έχουν πολλαπλά σημεία εισόδων, αλλά γενικά έχουν μόνο μια έξοδο. Το όνομα της κατάστασης γράφεται ως πρώτη είσοδος στο ορθογώνιο σχήμα καταστάσεων. Η ανάθεση του κώδικα της κατάστασης, εάν είναι γνωστή, γράφεται δίπλα στην ανώτερη δεξιά γωνία του σχήματος καταστάσεων. Η συνθήκη απόφασης είναι σε σχήμα ρόμβου που περιέχει είτε ένα σήμα εισόδου είτε μια λογική έκφραση. Δύο έξοδοι '0' ή '1' οδηγούνται σε μια άλλη συνθήκη απόφασης (όχι απαραίτητα), ένα ορθογώνιο ή σε έξοδο Mealy. Το στρογγυλεμένο οβάλ χρησιμοποιείται για τις εξόδους της Mealy. Ακολουθεί το σχήμα 4.5



Σχήμα 4.5 Διάγραμμα ροής 1

Στο επόμενο διάγραμμα ροής του σχήματος 4.6 μπορούμε να διακρίνουμε δύο λογικές εκφράσεις μέσα στην συνθήκη απόφασης. Στην πρώτη συνθήκη και όταν η απόφαση είναι True η μηχανή μεταβαίνει σε νέα κατάσταση, ενώ όταν η απόφαση επαληθεύεται ως False η μηχανή θα μεταβεί σε μια νέα λογική έκφραση και ανάλογα την συνθήκη που θα επικρατεί θα αλλάξει κατάσταση.



Σχήμα 4.6 Διάγραμμα ροής 2

4.4 Σύγκριση

Οι τρεις μέθοδοι αναπαράστασης ΜΠΚ, δηλαδή τα διαγράμματα καταστάσεων, οι πίνακες καταστάσεων καθώς και τα διαγράμματα ροής είναι όλα ισοδύναμα, δεδομένου ότι όλα περιγράφουν την ίδια δομή υλικού. Κάθε μέθοδος έχει τα δικά της ιδιαίτερα πλεονεκτήματα. Αν και το πιο δημοφιλές-το διάγραμμα μετάβασης καταστάσεων-, γίνεται πιο σύνθετο για προβλήματα όπου οι μεταβάσεις καταστάσεων εξαρτώνται από πολλές εισόδους, από την στιγμή που οι όροι μετάβασης γράφονται άμεσα στα βέλη μετάβασης. Αντίστοιχα ο πίνακας καταστάσεων, επιτρέπει στον σχεδιαστή τον αυστηρό έλεγχο των σημάτων λογικής. Τέλος, το διάγραμμα ροής είναι κατάλληλο για μικρά προβλήματα, όπου δεν υπάρχουν περισσότερες από περίπου 10 καταστάσεις και όπου μέχρι 2 ή 3 εισόδους ή εκφράσεις εισόδων, εξετάζονται μέσα σε κάθε κατάσταση. Για μεγαλύτερα προβλήματα, μπορεί να γίνουν αδέξια.

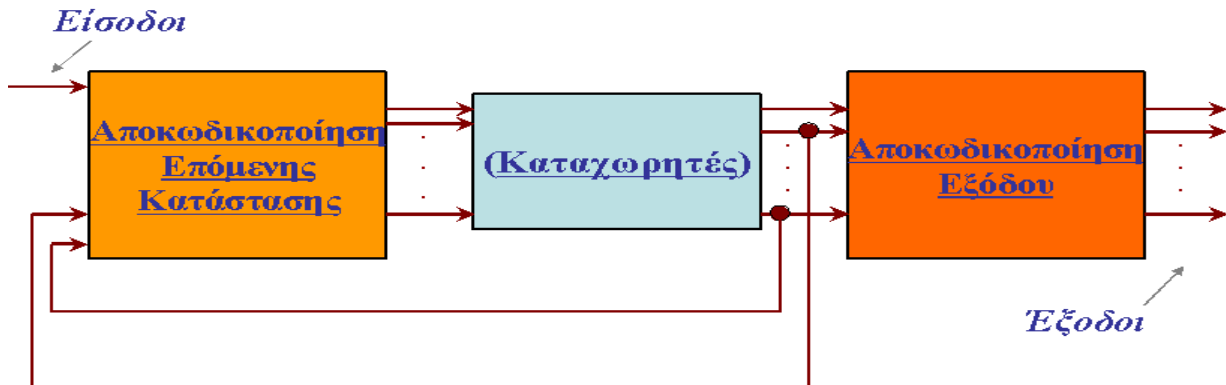
ΚΕΦΑΛΑΙΟ 5

Μηχανές Moore και Mealy

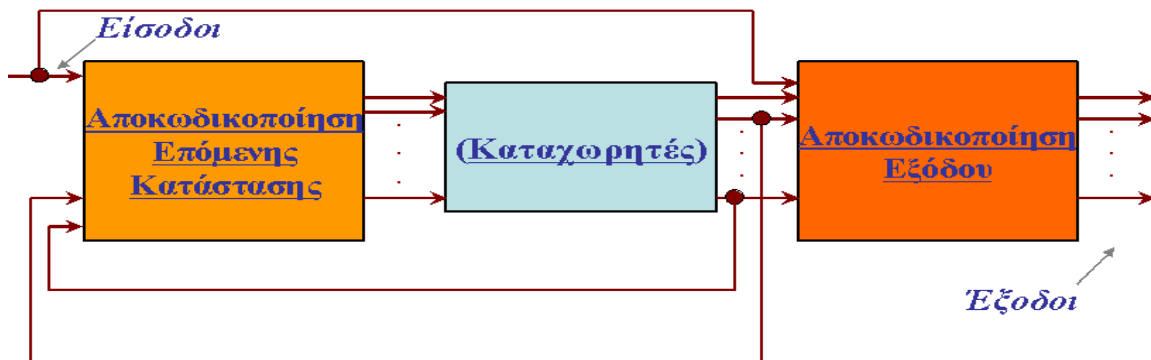
5.1 Εισαγωγή

Η γενική δομή μιας Μηχανής Πεπερασμένων Καταστάσεων χαρακτηρίζεται από τις παρούσες καταστάσεις (ΠΚ), τις επόμενες καταστάσεις (ΕΚ), και τις εξόδους (ΟΒ, ΟΑ), Αυτό θα μας βοηθήσει να επεξηγήσουμε πως οι μηχανές Moore (Μο) και Mealy (Με) εφαρμόζονται με τις περισσότερες ακολουθιακές συσκευές, που παρέχουν μια ενιαία συνδυαστική λογική και για τις επόμενες καταστάσεις και για τις λειτουργίες του αποκωδικοποιητή εξόδου. Υπάρχουν τέσσερις τρόποι που έχουμε τη δυνατότητα να χρησιμοποιήσουμε μια ακολουθιακή μηχανή, δύο από τους οποίους εφαρμόζουν οι μηχανές Μο και δύο οι μηχανές Με:

Ο υπολογισμός της εξόδου καθορίζει και τον τύπο μιας ΜΠΚ. Έτσι, υπάρχουν δύο ευρέως γνωστοί τύποι ΜΠΚ. Όσο αφορά την Μο, όπως φαίνεται και στο σχήμα 5.1 οι έξοδοι είναι λειτουργία αποκλειστικά και μόνο της παρούσας κατάστασης. Στην Με (σχήμα 5.2) οι έξοδοι είναι λειτουργία όχι μόνο της παρούσας κατάστασης, αλλά και των σημάτων εισόδου. Η λογική που απαιτείται είναι γνωστή ως λειτουργία εξόδου. Για οποιαδήποτε από τους δύο συγκεκριμένους τύπους μηχανής καταστάσεων, ο έλεγχος ακολουθίας εξαρτάται τόσο από τις καταστάσεις όσο και από τα σήματα εισόδου.



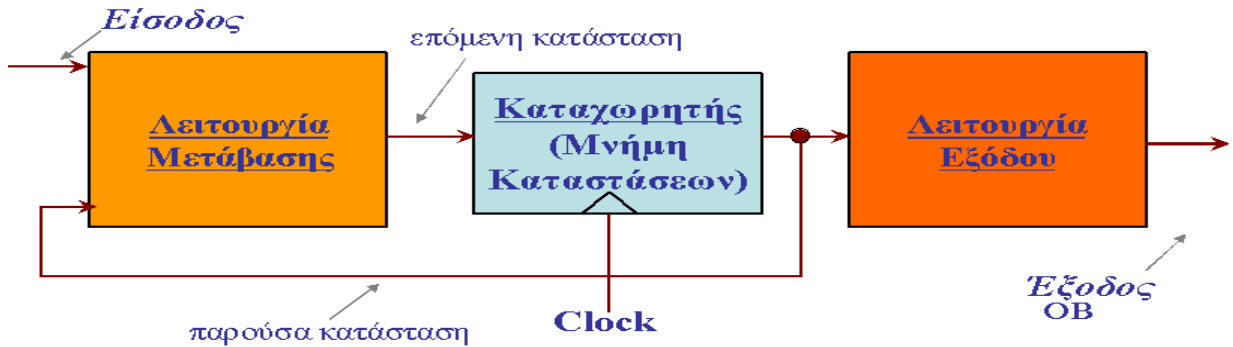
Σχήμα 5.1 Μηχανή Moore



Σχήμα 5.2 Μηχανή Mealy

5.2 Ασύγχρονη Moore

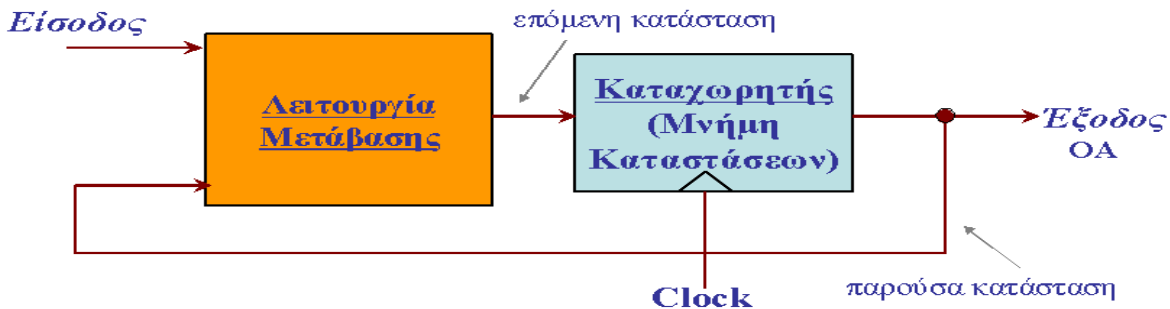
Οι έξοδοι (παρούσα κατάσταση) του καταχωρητή ανατροφοδοτούνται και καθορίζουν την παρούσα κατάσταση της μηχανής. Η συνδυαστική λογική υλοποιεί την λειτουργία μετάβασης, που παράγει την επόμενη κατάσταση και την λειτουργία εξόδου, όπου παράγει την έξοδο της μηχανής. Επίσης, διακρίνουμε ότι η έξοδος εξαρτάται αποκλειστικά και μόνο από την παρούσα κατάσταση της μηχανής (σχήμα 5.3).



Σχήμα 5.3 Ασύγχρονη μηχανή Moore

5.3 Σύγχρονη Moore

Η συνδυαστική λογική μπορεί να είναι η λειτουργία ενότητας. Η έξοδος παράγεται άμεσα μαζί με την παρούσα κατάσταση, όπως διακρίνουμε και στο σχήμα 5.4



Σχήμα 5.4 Σύγχρονη μηχανή Moore

5.4 Ασύγχρονη Mealy

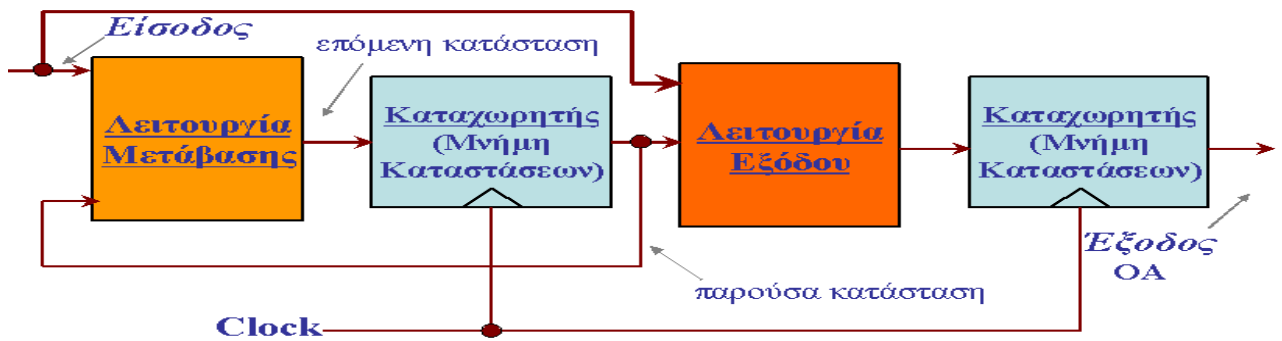
Οι έξοδοι (παρούσα κατάσταση) του καταχωρητή ανατροφοδοτούνται και καθορίζουν την παρούσα κατάσταση της μηχανής. Η συνδυαστική λογική υλοποιεί την λειτουργία μετάβασης, όπου παράγει την επόμενη κατάσταση και την λειτουργία εξόδου, όπου παράγει την έξοδο της μηχανής. Επίσης, διακρίνουμε ότι η έξοδος εξαρτάται και από την παρούσα κατάσταση και από την είσοδο. Ακολουθεί το αντίστοιχο σχήμα



Σχήμα 5.5 Ασύγχρονη μηχανή Mealy

5.5 Σύγχρονη Mealy

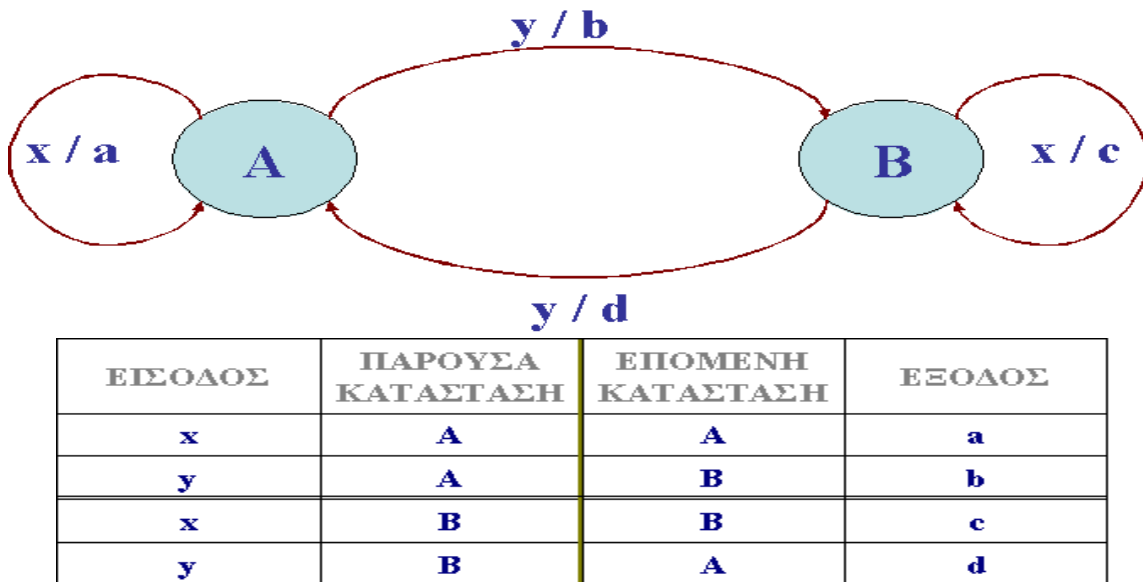
Μια εναλλακτική μορφή της μηχανής Me (σχήμα 5.6), όπου οι έξοδοι διαβιβάζονται μέσω ενός επιπλέον καταχωρητή εξόδου και αυτό έχει σαν αποτέλεσμα, να μην ανταποκρίνεται αμέσως στις αλλαγές της εισόδου. Επίσης, διακρίνουμε ότι οι δύο καταχωρητές συγχρονίζονται από ένα ενιαίο ρολόι.



Σχήμα 5.6 Σύγχρονη μηχανή Mealy

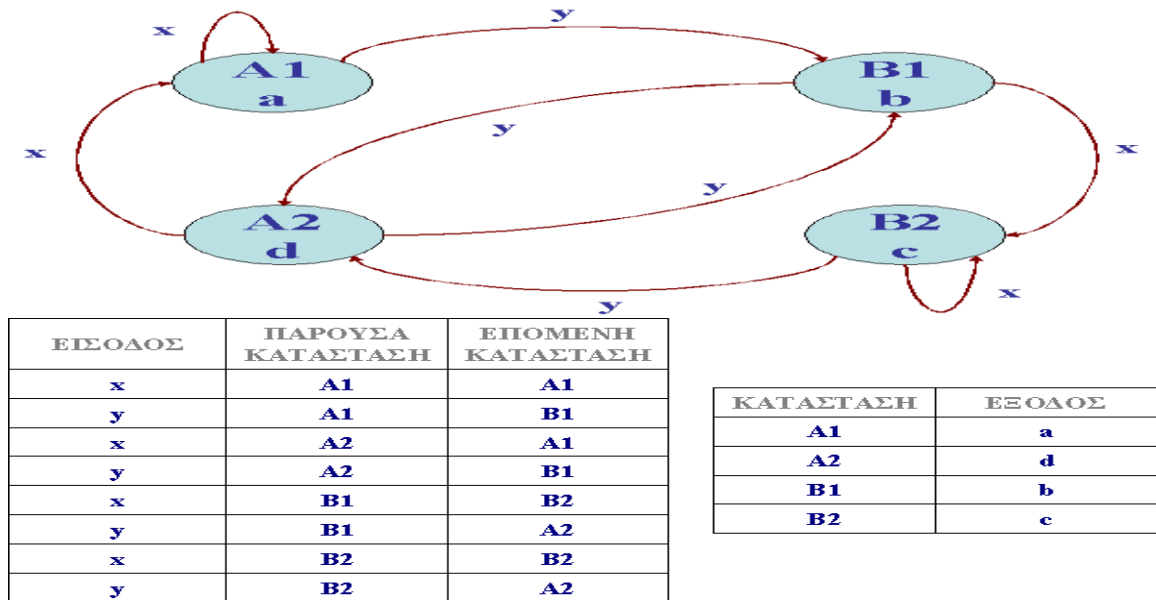
5.6 Σύγκριση μηχανών Moore / Mealy

Ένα θεώρημα της θεωρίας αυτομάτων πεπερασμένων καταστάσεων ότι **οποιαδήποτε μηχανή Moore μπορεί να μετατραπεί σε μια Mealy μηχανή, και αντίστροφα**. Σαφώς, εάν μια μηχανή Moore έχει εξόδους που εξαρτώνται μόνο από την κατάσταση, και η μηχανή έχει n ξεχωριστές τιμές εξόδου, πρέπει να έχει n καταστάσεις. Κατά συνέπεια, η μετάφραση από Mealy σε Moore οδηγεί συχνά στην προσθήκη καταστάσεων! Αυτό το θεώρημα θα το δούμε αναλυτικά στη συνέχεια της ενότητας. Ο ακόλουθος πίνακας καταστάσεων και η εικόνα καθορίζει μια Mealy μηχανή.



Σχήμα 5.7 Διάγραμμα και Πίνακας μετάβασης καταστάσεων μηχανής Mealy

Στο διάγραμμα καταστάσεων για μια Mealy μηχανή, κάθε τόξο (μετάβαση κατάστασης) ονομάζεται με την είσοδο που επιτρέπει εκείνη την μετάβαση, μια κάθετο, και την έξοδο που συνδέεται με εκείνη την μετάβαση. Η έξοδος είναι πάντα μετά από ή κάτω από την κάθετο. Η ανωτέρω μηχανή είναι ισοδύναμη με την ακόλουθη μηχανή Moore:



Σχήμα 5.8 Διάγραμμα και Πίνακας μετάβασης καταστάσεων μηχανής Moore

Σημειώνουμε ότι η περιγραφή μιας μηχανής Mo έχει συχνά δύο πίνακες, έναν για τις μεταβάσεις καταστάσεων, και έναν δίνοντας την έξοδο που συνδέεται με κάθε κατάσταση. Στα ανωτέρω, οι καταστάσεις που ήταν χωρισμένες έχουν διατηρήσει τα παλαιά ονόματά τους, αλλά παρατηρούμε ότι υπάρχουν δύο περισσότεροι κύκλοι για να διακρίνουμε τις δύο καταστάσεις Mo ως αποτέλεσμα κάθε Me κατάστασης. Η έξοδος που συνδέεται με κάθε κατάσταση σε μια μηχανή Mo γράφεται μέσα στον κύκλο για εκείνη την κατάσταση, κάτω από το όνομα της κατάστασης.

Δυαδικές κωδικοποιήσεις

Η ανωτέρω περιγραφή Moore και των Mealy μηχανών χρησιμοποίησε συμβολικά ονόματα για τις εισόδους και τις εξόδους. Σαφώς, οι δυαδικοί κώδικες μπορούν να χρησιμοποιηθούν για κάθε έναν. Έτσι, παραδείγματος χάριν, θα μπορούσαμε να κωδικοποιήσουμε τις εισόδους και τις εξόδους που χρησιμοποιήθηκαν στα ανωτέρω παραδείγματα όπως:

ΕΙΣΟΔΟΣ	ΚΩΔΙΚΟΠΟΙΗΣΗ
x	0
y	1

ΕΞΟΔΟΣ	ΚΩΔΙΚΟΠΟΙΗΣΗ
a	00
b	01
c	10
d	11

Πίνακας 4 Κωδικοποίηση δεδομένων

Ομοίως, μπορούμε να κωδικοποιήσουμε τα ονόματα των καταστάσεων:

MEALY	ΚΩΔΙΚΟΠΟΙΗΣΗ
A	0
B	1

MOORE	ΚΩΔΙΚΟΠΟΙΗΣΗ
A1	00
A2	01
B1	10
B2	11

Πίνακας 5 Κωδικοποίηση καταστάσεων

Κάνοντας αυτές τις αναθέσεις καταστάσεων, μπορούμε να μετατρέψουμε τους πίνακες μεταβάσεων και τον πίνακα εξόδου που δίνονται προηγουμένως για να δώσει τα εξής:

<u>Μηχανή Mealy</u>			
ΕΙΣΟΔΟΣ	ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ
0	0	0	00
1	0	1	01
0	1	1	10
1	1	0	11

<u>Μηχανή Moore</u>		
ΕΙΣΟΔΟΣ	ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ
0	00	00
1	00	10
0	01	00
1	01	10
0	10	11
1	10	01
0	11	11
1	11	01

ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ
00	00
01	11
10	01
11	10

Πίνακας 6 Κωδικοποίηση μηχανών Moore και Mealy

Στη σύγκριση του Mo και των Me μηχανών, σημειώνουμε ότι ο αριθμός των bits στην επόμενη-κατάσταση εξόδου θα είναι συνήθως περισσότερος για μια μηχανή Moore απ' ότι για μια ισοδύναμη Mealy μηχανή, αλλά αυτό, όπου η μηχανή Moore χρειάζεται μόνο τον καταχωρητή καταστάσεων για να αποθηκεύσει την επικρατούσα κατάσταση, η Mealy μηχανή πρέπει να αποθηκεύσει τόσο την επικρατούσα κατάσταση όσο και την τρέχουσα έξοδο.

Αυτή η απαίτηση ενός καταχωρητή να κρατηθεί η τρέχουσα έξοδος επιβάλλεται στις Me μηχανές για να αποτρέψει την έξοδο από την αλλαγή εκτός από όταν φθάνει η άκρη ρολογιού, δείχνοντας ότι η μηχανή πρέπει να προωθήσει στην επόμενη κατάσταση. Εάν αυτό δεν γίνει, η έξοδος της Mealy μηχανής θα μπορούσε να αλλάξει, οπότε οποιαδήποτε είσοδο στη μηχανή αλλάζει, και αυτό είναι γενικά ανεπιθύμητο. Τα παραδείγματα που δόθηκαν ανωτέρω ήταν σκόπιμα, αδέξια εν προκειμένω.

ΚΕΦΑΛΑΙΟ 6

Αλγόριθμος Σχεδίασης Μηχανών Πεπερασμένων Καταστάσεων

6.1 Βασικά βήματα σχεδίασης

Η διαδικασία μελέτης και σχεδίασης των Μηχανών Πεπερασμένων Καταστάσεων συνοψίζεται σε έξι βασικά στάδια:

Στάδιο 1^ο: Κατανόηση των δηλώσεων των προδιαγραφών μιας ειδικής εφαρμογής. Συνήθως, βοηθάει τον σχεδιαστή αναπαριστώντας την εφαρμογή που έχει να υλοποιήσει σχεδιάζοντας μια εικόνα.

Στάδιο 2^ο: Παρουσίαση της ΜΠΚ με τους τρόπους αναπαράστασης που παρουσιάστηκαν προηγουμένως. Όπως για τον σχεδιασμό ενός ακολουθιακού κυκλώματος ήταν καταρχήν απαραίτητο η κατασκευή του πίνακα αληθείας εκ των δεδομένων του προβλήματος, έτσι και στις ΜΠΚ κατασκευάζεται αρχικά ο πίνακας καταστάσεων. Σε αυτόν εμφανίζονται οι παρούσες καταστάσεις που θα λάβει το κύκλωμα για κάθε μεταβολή των εισόδων του και η έξοδος του κυκλώματος για κάθε μεταβολή της εισόδου. Για την κατασκευή του πίνακα καταστάσεων πολύπλοκων ακολουθιακών κυκλωμάτων οδηγούμεθα ευκολότερα εάν κατασκευάσουμε πρώτα το διάγραμμα καταστάσεων, στο οποίο φαίνεται περισσότερο παραστατικά η όλη περιγραφή του υπό μελέτη προβλήματος.

Στάδιο 3^ο: Ελαχιστοποίηση καταστάσεων, δηλαδή η ανεύρεση των ομοίων καταστάσεων και συγχώνευσή τους. Πιθανόν σε έναν πίνακα καταστάσεων να υπάρχουν πολλές όμοιες καταστάσεις, οι οποίες ως εκ τούτου είναι πλεονασμός και θα πρέπει να συγχωνευθούν σε μία. Αυτό φυσικά επιδρά στην οικονομικότερη πραγματοποίηση του κυκλώματος. Δύο ή περισσότερες καταστάσεις είναι όμοιες, όταν έχουν για όλες τις ακολουθίες εισόδων τις ίδιες επόμενες καταστάσεις αλλά και την ίδια έξοδο. Οι καταστάσεις αυτές μπορούν να συγχωνευθούν σε μία. Μετά τον εντοπισμό των ομοίων καταστάσεων και την συγχώνευσή τους, κατασκευάζουμε τον τελικό πίνακα καταστάσεων, βάσει του οποίου δύναται να κατασκευασθεί το τελικό διάγραμμα καταστάσεων.

Στάδιο 4^ο: Κωδικοποίηση των δεδομένων της εφαρμογής. Αφού κατασκευασθεί ο τελικός πίνακας καταστάσεων, πρέπει όλες οι καταστάσεις του πίνακα αυτού να κωδικοποιηθούν βάση ενός δυαδικού κώδικα. Ο *BCD 8421* είναι ο πιο συνηθισμένος κώδικας και για αυτό ονομάζεται και φυσικός δυαδικός κώδικας. Επίσης, σε πολλές περιπτώσεις εφαρμογών, όπως για παράδειγμα στη μετατροπή αναλογικών σημάτων σε ψηφιακά, επιθυμείται η χρησιμοποίηση δυαδικού κώδικα του οποίου οι παραστάσεις διαδοχικών δεκαδικών ψηφίων να διαφέρουν μεταξύ τους μόνο κατά ένα δυφίο, δηλαδή οι δυαδικοί συνδυασμοί να είναι γειτονικοί. Οι κώδικες που έχουν αυτή την ιδιότητα αναφέρονται ως κυκλικοί κώδικες. Ο σπουδαιότερος απ' αυτούς είναι ο κώδικας *Gray*.

Γενικά, υπάρχουν πολλές εκχωρήσεις για την κωδικοποίηση των καταστάσεων. Η εκλογή του καταλληλότερου κώδικα θα έχει ως συνέπεια ένα οικονομικότερο κύκλωμα, αφού διαφορετικές καταστάσεις οδηγούν σε διαφορετικές εξισώσεις και κατά συνέπεια σε διαφορετικά ισοδύναμα κυκλώματα. Δυστυχώς, συστηματική μέθοδος για την ανεύρεση του καταλληλότερου κώδικα δεν υπάρχει και πολλές φορές γίνεται εκ πείρας και τύχη του σχεδιαστή. Ωστόσο, υπάρχουν διάφοροι μέθοδοι και κανόνες, οι οποίοι υποβοηθούν για την εύρεση του κατάλληλου κώδικα, χωρίς όμως αυτό να σημαίνει ότι είναι και ο άριστος.

Στάδιο 5^ο: Επιλογή τύπου φλιπ-φλοπ για την υλοποίηση του καταχωρητή καταστάσεων της ΜΠΚ. Χρησιμοποιώντας τους πίνακες Καρνώ και αφού έχει εκλεγεί και ο τύπος των χρησιμοποιηθησόμενων φ-φ (εκλογή η οποία επηρεάζει το τελικό κύκλωμα) εξάγονται οι τελικές εξισώσεις του κυκλώματος εισόδου, των φ-φ και του λογικού κυκλώματος εξόδου.

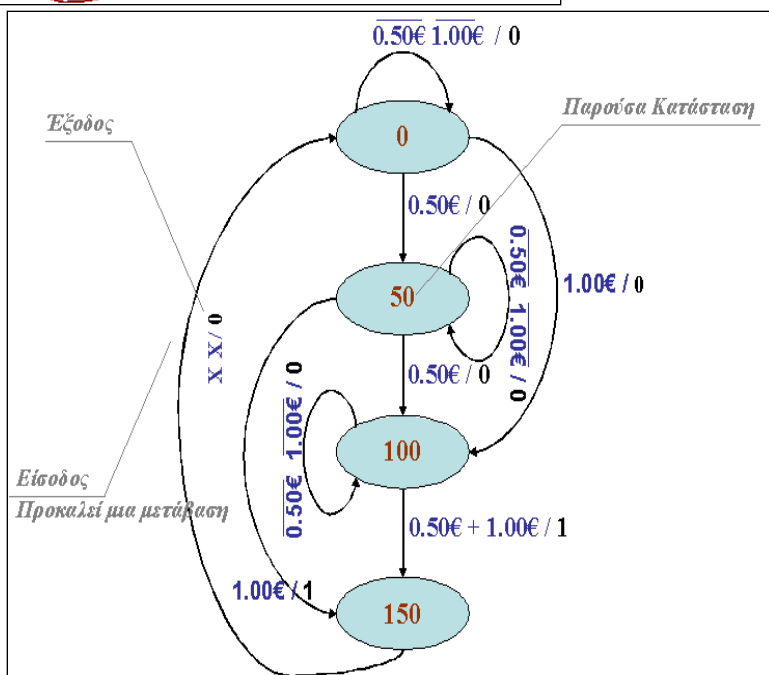
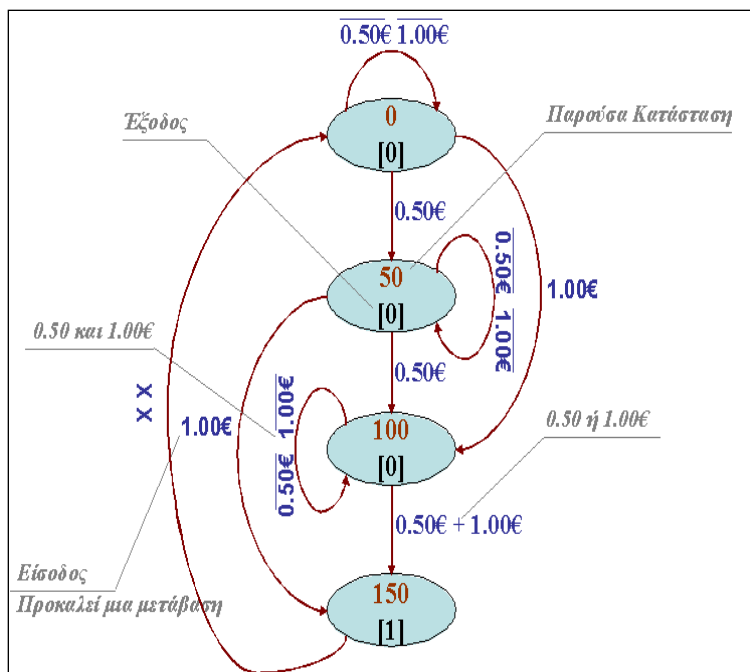
Στάδιο 6^ο: Η τελική υλοποίηση του ακολουθιακού κυκλώματος.

ΚΕΦΑΛΑΙΟ 7

Γενικές εφαρμογές

7.1 Μηχανή Πώλησης : Είναι μια μηχανή που συναντάμε σχεδόν καθημερινά στη ζωή μας. Είναι μία συσκευή υπολογισμού σε λειτουργία και έχουμε τη δυνατότητα να τη μοντελοποιήσουμε ως Μηχανή Πεπερασμένων Καταστάσεων. Στη γενική μορφή της, μια μηχανή πώλησης μπορεί να έχει εισόδους, όπως κέρματα ή κουμπιά τα οποία επιτελούν διάφορες λειτουργίες καθώς και εξόδους, οι οποίες δύναται να είναι φαγητό, κουτί αναψυκτικών, εισιτήρια αλλά και κέρματα (ρέστα).

Η εφαρμογή που περιγράφουμε προσδιορίζει μια μηχανή πώλησης η οποία δέχεται ένα καθορισμένο χρηματικό ποσό και εξάγει ένα εισιτήριο. Παίρνει κέρματα των 0.50€ και 1.00€ και πουλάει το εισιτήριο που κοστίζει 1.50€. Εάν πληρώσεις δύο κέρματα των 1.00€, κρατάει τα ρέστα. Οι μεταβάσεις πραγματοποιούνται όταν εισάγεις στη μηχανή ένα κέρμα. Από την στιγμή που η μηχανή θα παράγει το εισιτήριο, οποιαδήποτε είσοδο λάβει, θα μεταβεί την αρχική της κατάσταση, ώστε να ξεκινήσει ένα νέο κύκλο επεξεργασίας των δεδομένων. Την εφαρμογή μας την υλοποιούμε και με τους δύο τύπους των ΜΠΚ (Moore και Mealy). Βέβαια στο συγκεκριμένο παράδειγμα εφαρμόζουμε κάποια βήματα της σχεδίασης των ΜΠΚ και δεν αναλύουμε όλη την διαδικασία.



Σχήμα 7.1 Διάγραμμα μετάβασης καταστάσεων (Me/Mo) μηχανής πώλησης

Καταστάσεις: Περιγραφή του εσωτερικού περιβάλλοντος της μηχανής. Έχουμε ορίσει ως εσωτερικές καταστάσεις (πεπερασμένες) στο παράδειγμα μας τις εξής: 0, 50, 100

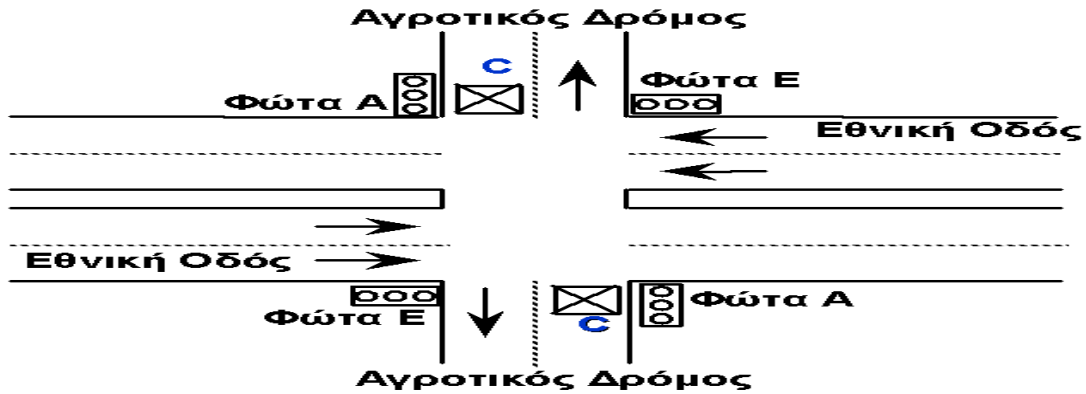
Είσοδοι: Κέρματα των 0.50€ και 1.00€.

Έξοδοι: Εισιτήριο.

Κανόνες: Εδώ προσδιορίζουμε την αντίδραση της μηχανής μας στα διάφορα γεγονότα. Οι βασικοί μας κανόνες είναι ότι η μηχανή θα εξάγει το εισιτήριο όταν δεχτεί 1.50€, ενώ κάθε φορά που θα βγάζει το εισιτήριο θα μεταβαίνει στην αρχική κατάσταση (Reset της μηχανής).

7.2 Ελεγκτής φωτός κυκλοφορίας

Μια πολυάσχολη εθνική οδός διασταυρώνεται από το λίγο χρησιμοποιημένο αγροτικό δρόμο.



Σχήμα 7.2

Εικόνα ελεγκτή φωτός κυκλοφορίας

Υπάρχουν ανιχνευτές που τοποθετούνται κατά μήκος του αγροτικού δρόμου για να ανιχνεύουν την παρουσία οχήματος. Ο ελεγκτής φωτός κυκλοφορίας πρέπει να λειτουργεί ως εξής: Εφόσον δεν ανιχνεύεται κανένα όχημα στον αγροτικό δρόμο, τα φώτα στην εθνική οδό πρέπει να παραμείνουν πράσινα. Εάν ανιχνεύεται ένα όχημα στον αγροτικό δρόμο, τα φώτα στην εθνική οδό πρέπει να αλλάξουν από πράσινο σε πορτοκαλί σε κόκκινο, που επιτρέπει πλέον στα φώτα του αγροτικού δρόμου να γίνουν πράσινα. Τα φώτα στον αγροτικό δρόμο μένουν πράσινα μόνο εφόσον ανιχνεύεται αμάξι και ποτέ περισσότερο από ένα προκαθορισμένο διάστημα, για να επιτρέψει στην κυκλοφορία να ρέει κατά μήκος της εθνικής οδού. Εάν ικανοποιούνται αυτοί οι όροι, τα φώτα στον αγροτικό δρόμο αλλάζουν από πράσινο σε πορτοκαλί στο κόκκινο, επιτρέποντας έτσι στα φώτα στην εθνική οδό για να επιστρέψουν σε πράσινο. Τέλος, ακόμα και αν ανιχνεύονται οχήματα στον αγροτικό δρόμο, η εθνική πρέπει να παραμείνει πράσινη για ένα καθορισμένο διάστημα.

ΚΕΦΑΛΑΙΟ 8 **ΔΙΚΤΥΑ ΕΠΙΚΟΙΝΩΝΙΩΝ**

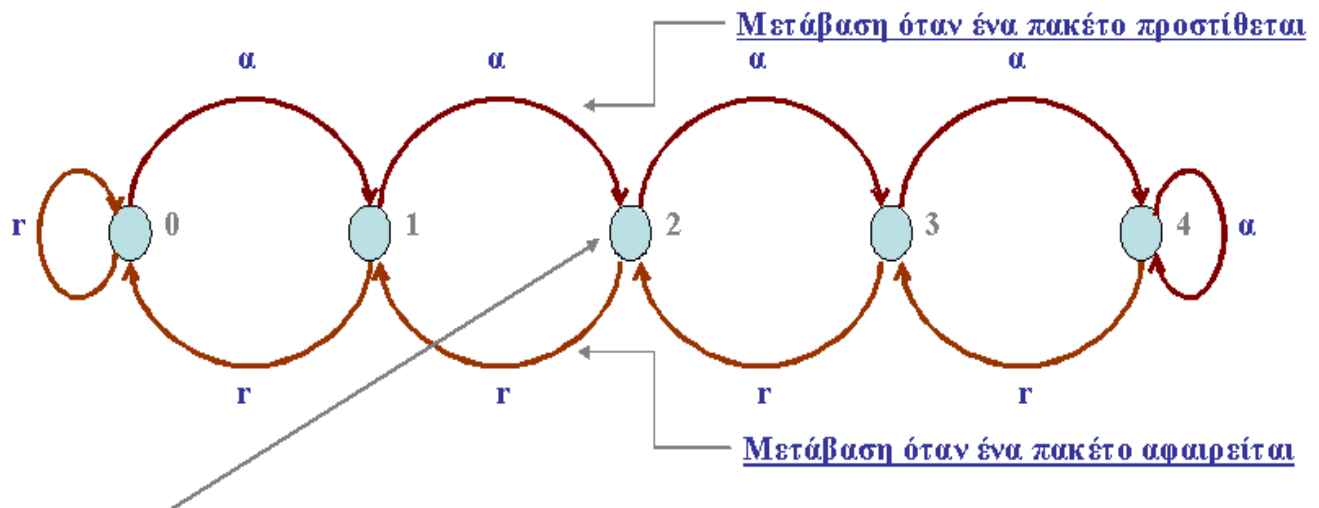
8.1 Αναπαράσταση ενός πρωτοκόλλου

Μια Μηχανή Πεπερασμένων Καταστάσεων μπορεί να αναπαραστήσει ένα απλό πρωτόκολλο ζεύξης δεδομένων και να δείξει ότι το πρωτόκολλο αυτό δεν είναι σωστό. Το να επαληθεύσουμε ένα πρωτόκολλο σημαίνει να διαπιστώσουμε ότι έχει την επιθυμητή συμπεριφορά. Μια μέθοδος αυτόματης επαλήθευσης πρωτοκόλλων είναι και με ΜΠΚ. Όπως έχουμε προαναφέρει, μια ΜΠΚ περιγράφει πώς οι είσοδοι μεταβάλλουν

την κατάσταση. Όταν οριστεί ένα πρωτόκολλο, μπορεί να κατασκευαστεί ένα μοντέλο ΜΠΚ της επικοινωνιακής ζεύξης και μέσω αυτού μπορούν να ανιχνευθούν μη αποδεκτές ακολουθίες καταστάσεων της ΜΠΚ.

Μια ΜΠΚ είναι το μοντέλο της εξέλιξης μιας συσκευής (μηχανής), η οποία υπόκειται σε εξωτερικές εισόδους. Η μηχανή μπορεί να βρίσκεται σε ένα πεπερασμένο πλήθος από διαφορετικές καταστάσεις. Η συσκευή μπορεί να είναι ένας καταχωρητής, ο οποίος μπορεί να κρατάει τέσσερα πακέτα, και η κατάσταση μπορεί να είναι 0, 1, 2, 3 ή 4 για την αναπαράσταση του αριθμού των πακέτων στην μνήμη. Το σύνολο των πιθανών καταστάσεων της ΜΠΚ, ο χώρος καταστάσεων, δηλώνεται με X .

Η ΜΠΚ υπόκειται σε εξωτερικές εισόδους : πακέτα μπορεί να προστίθενται ή να αφαιρούνται από την μνήμη. Οι εισοδοί εφαρμόζονται μία κάθε φορά και παίρνουν τιμές από ένα πεπερασμένο σύνολο U . Στο σχήμα 8.1 του καταχωρητή, το σύνολο των τιμών της εισόδου είναι $U = \{a, r\}$, όπου το a σημαίνει 'προσθέτω ένα πακέτο' και r 'αφαιρώ ένα πακέτο'. Οι εισοδοί μεταβάλλουν την κατάσταση της ΜΠΚ.



Η κατάσταση αυτή υποδηλώνει ότι υπάρχουν δύο πακέτα στον καταχωρητή

Σχήμα 8.1 Διάγραμμα μεταβολής των καταστάσεων για καταχωρητή τεσσάρων πακέτων

Στο παράδειγμα της μνήμης, εάν η είσοδος είναι a όταν η κατάσταση είναι 2, τότε η κατάσταση γίνεται 3. Η νέα κατάσταση εξαρτάται από την προηγούμενη κατάσταση και από την είσοδο. Γενικά, εάν η είσοδος με τιμή $u \in U$ εφαρμοστεί ενώ η μηχανή είναι στην κατάσταση $x \in X$, τότε η μηχανή μεταπηδάει στην νέα κατάσταση $g(x,u)$ όπου $g: X * U \rightarrow X$ είναι μια δοσμένη συνάρτηση, που ονομάζεται συνάρτηση μεταβολής καταστάσεων. Επομένως, στο παράδειγμα του καταχωρητή, έχουμε $g(2,a) = 3$. Για τον καθορισμό της ακολουθίας των καταστάσεων η οποία προκύπτει από μια δοσμένη ακολουθία εισόδων, αρκεί να γνωρίζουμε την συνάρτηση g και την αρχική κατάσταση x_0 .

Το διάγραμμα μεταβολής καταστάσεων αποδίδει περιληπτικά τον καθορισμό της ΜΠΚ. Αναπαριστά όλες τις καταστάσεις της ΜΠΚ μαζί με βέλη μεταξύ των καταστάσεων τα οποία αναπαριστούν τις πιθανές μεταβολές. Το σχήμα δείχνει το διάγραμμα μεταβολής καταστάσεων για τον καταχωρητή που χωράει τέσσερα πακέτα. Στη συνέχεια θα δούμε πως μπορεί να μοντελοποιηθεί η εξέλιξη ενός πρωτοκόλλου με ΜΠΚ. Θεωρούμε ένα πρωτόκολλο με μη αριθμημένα πακέτα και ACKs. Θα

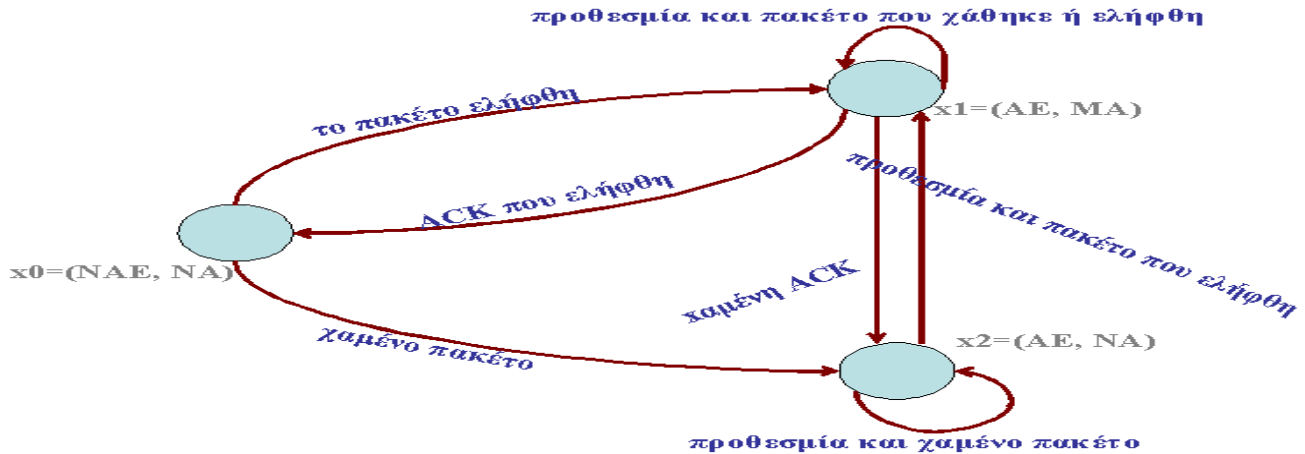
μοντελοποιήσουμε αυτό το πρωτόκολλο με μια ΜΠΚ, και θα δείξουμε ότι δεν μπορεί να παραδώσει τα πακέτα στη σωστή σειρά και χωρίς λάθος.

Το πρώτο βήμα είναι να αναγνωρίσουμε την συσκευή και τις καταστάσεις της. Η συσκευή αποτελείται από έναν αποστολέα, έναν παραλήπτη και μια επικοινωνιακή ζεύξη από τον παραλήπτη στον αποστολέα. Θα καθορίσουμε καταστάσεις κατάλληλες για την αναπαράσταση των βημάτων της εκτέλεσης του πρωτοκόλλου. Για παράδειγμα, δεν θα ξεχωρίσουμε αν έχει γίνει η μετάδοση ενός πακέτου από τον αποστολέα πριν 3ms ή πριν 5ms. Στην πραγματικότητα, θα εξετάσουμε τις καταστάσεις της συσκευής μόνο στο τέλος της διάδοσης πακέτων και επιβεβαιώσεων. Για λόγους απλότητας, η επικοινωνιακή ζεύξη θεωρείται ότι είναι πλήρως αμφίδρομη. Με AE (Acknowledgement Expected, αναμένεται επιβεβαίωση) δηλώνουμε την κατάσταση του αποστολέα στην οποία περιμένει μια ACK και με NAE (No Acknowledgement Expected, δεν αναμένεται επιβεβαίωση) την κατάσταση στην οποία δεν περιμένει ACK.

Παρόμοια, δηλώνουμε με MA (Must Acknowledge, χρειάζεται επιβεβαίωση) την κατάσταση του παραλήπτη κατά την οποία πρέπει να επιβεβαιώσει ένα πακέτο που έλαβε και με NA (No Acknowledgement, δεν χρειάζεται επιβεβαίωση) την κατάσταση κατά την οποία δεν χρειάζεται να σταλεί ACK. Όσο είναι στην κατάσταση MA ο παραλήπτης μπορεί να λάβει ένα πακέτο. Αυτό συμβαίνει όταν ο παραλήπτης περιμένει για αρκετή ώρα πριν να σταλεί κάποια επιβεβαίωση. Μία δυνατή κατάσταση δηλώνεται με $x = (s, r)$ όπου **se** [AE, NAE] είναι η κατάσταση του αποστολέα και **re** [MA, NA] είναι η κατάσταση του παραλήπτη.

Το δεύτερο βήμα είναι ο καθορισμός των μεταβάσεων από κατάσταση σε κατάσταση. Για να γίνει αυτό, πρέπει να αναγνωρίσουμε όλες τις πιθανές μεταβάσεις από οποιαδήποτε δοσμένη κατάσταση. Οι μεταβάσεις προκαλούνται από ορισμένες εισόδους. Για παράδειγμα, ας πούμε ότι $x = x_0 := (NAE, NA)$: ο αποστολέας δεν περιμένει ACK και ο παραλήπτης δεν έχει πακέτο προς επιβεβαίωση. Αυτή είναι η κατάσταση πριν την μετάδοση του πρώτου πακέτου, όταν το σύστημα ξεκινάει.

Η κατάσταση αλλάζει μόλις ο αποστολέας μεταδώσει ένα πακέτο. Δύο είναι οι πιθανές εξελίξεις: είτε το πακέτο φτάνει σωστά στον παραλήπτη ή χάνεται εξαιτίας λαθών μετάδοσης. Στην πρώτη περίπτωση, η επόμενη κατάσταση στο τέλος της διάδοσης του πακέτου είναι $x_1 := (AE, MA)$. Η κατάσταση σημαίνει ότι ο αποστολέας περιμένει μία ACK για ένα πακέτο που μετέδωσε και ότι ο παραλήπτης πρέπει να επιβεβαιώσει το πακέτο που μόλις έλαβε. Στη δεύτερη περίπτωση, η επόμενη κατάσταση είναι $x_2 := (AE, NA)$ καθώς ο παραλήπτης δεν επιβεβαιώνει τα λανθασμένα πακέτα. Εξετάζοντας όλες τις πιθανές μεταβάσεις από τις καταστάσεις x_1 και x_2 , φτάνουμε στο διάγραμμα μεταβάσεων του σχήματος 8.2, όπου φαίνονται τα γεγονότα που αντιπροσωπεύει κάθε βέλος.



Σχήμα 8.2 Διάγραμμα μετάβασης καταστάσεων για μη αριθμημένα πακέτα

Όταν δεν υπάρχει λάθος μετάβασης, η ακολουθία των γεγονότων είναι:

$X_0, X_1, X_0, X_1, \dots$

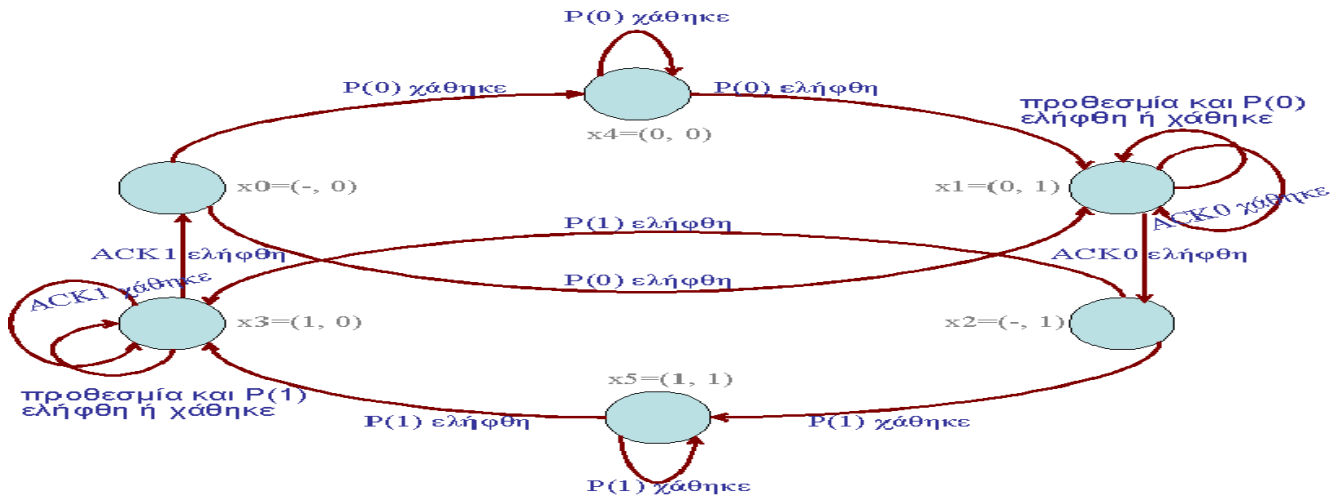
Υποστηρίζουμε ότι η επόμενη ακολουθία γεγονότων αντιπροσωπεύει μη αποδεκτή εκτέλεση του πρωτοκόλλου:

X_0, X_1, X_2, X_1

Αυτή είναι η ακολουθία των καταστάσεων όταν το πρώτο πακέτο φτάσει σωστά στον παραλήπτη ($x_0 \rightarrow x_1$), η επιβεβαίωση αυτού αλλοιωθεί ($x_1 \rightarrow x_2$), ο αποστολέας ξαναμεταδώσει ένα αντίγραφο του πακέτου μετά την λήξη της σχετικής προθεσμίας, και το αντίγραφο φτάσει σωστά στον παραλήπτη ($x_2 \rightarrow x_1$). Η ακολουθία είναι μη αποδεκτή επειδή ο παραλήπτης λαμβάνει δύο σωστά πακέτα-που το δεύτερο είναι αντίγραφο του πρώτου-χωρίς ο αποστολέας να λάβει ACK. Ο παραλήπτης δεν γνωρίζει ότι το δεύτερο πακέτο είναι αντίγραφο και θα νομίσει ότι πρόκειται για νέο πακέτο. Στο παράδειγμα αυτό, η παράσταση ως ΜΠΚ ήταν χρήσιμη για την εύρεση όλων των δυνατών ακολουθιών γεγονότων όταν το πρωτόκολλο εκτελείται.

8.2 Επαλήθευση της ορθότητας του ABP

Τώρα, θα εφαρμόσουμε την προσέγγιση της ΜΠΚ στην επαλήθευση του ABP (Alternating Bit Protocol, πρωτόκολλο εναλλασσομένου bit). Να επισημάνουμε ότι τα πακέτα και οι επιβεβαιώσεις αυτών αριθμούνται με 0 και 1. Υποθέτουμε ότι η ζεύξη είναι ημι-αμφίδρομη (half duplex) και εξετάζουμε, όπως και πριν, την κατάσταση της συσκευής στο τέλος της διάδοσης των πακέτων και των ACKs. Η κατάσταση του αποστολέα είναι είτε 0, 1 ή -. Η κατάσταση 0 αντιπροσωπεύει την αναμονή του αποστολέα για την ACK0. Η κατάσταση είναι 1 όταν ο αποστολέας περιμένει για την ACK1, και είναι - όταν ο αποστολέας δεν περιμένει ACK, δηλαδή όταν δεν εκκρεμούν ανεπιβεβαίωτα πακέτα. Η κατάσταση του παραλήπτη είναι 0 όταν περιμένει πακέτο με τον αριθμό 0. Η κατάσταση είναι 1 όταν περιμένει πακέτο με τον αριθμό 1. Η κατάσταση του πρωτοκόλλου είναι $x = (s, r)$ όπου s είναι η κατάσταση του αποστολέα και r είναι η κατάσταση του παραλήπτη. Το πρωτόκολλο υποθέτουμε ότι αρχικά είναι στην κατάσταση $x_0 := (-, 0)$. Το διάγραμμα καταστάσεων φαίνεται στο σχήμα 8.3.



Σχήμα 8.3 Διάγραμμα μεταβολής καταστάσεων του ABR

Όταν δεν υπάρχει λάθος μετάδοσης, η ακολουθία των καταστάσεων του πρωτοκόλλου είναι

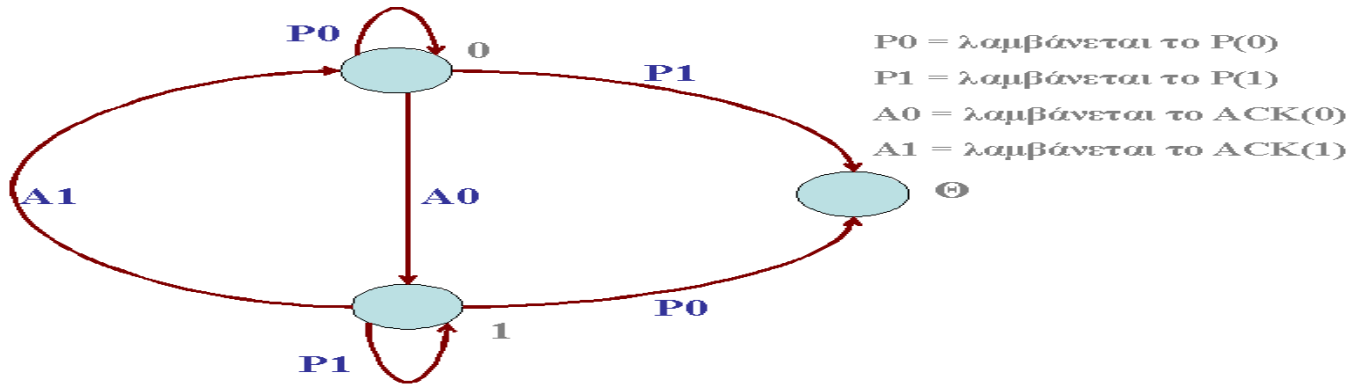
$$(-, 0) \rightarrow (0, 1) \rightarrow (-, 1) \rightarrow (1, 0) \rightarrow (-, 0) \rightarrow (0, 1) \rightarrow \dots$$

Οι μεταβάσεις αυτές αντιπροσωπεύουν την άφιξη του πακέτου 0, την άφιξη της ACK0, την άφιξη του πακέτου 1, την άφιξη της ACK1, κ.ο.κ. Εφόσον επαληθευθεί το **σχήμα**, η ορθότητα του ABP αποδεικνύεται. Πραγματικά, το **σχήμα** αυτό αποδεικνύει, ακόμα και με αυθαίρετα λάθη μετάδοσης, ότι ο παραλήπτης λαμβάνει ένα πακέτο με αριθμό 0, έπειτα ο αποστολέας παίρνει ACK0, έπειτα ο παραλήπτης λαμβάνει ένα πακέτο με αριθμό 1, κ.ο.κ.

8.3 Μεταφορά του διαγράμματος σε ηλεκτρονικό υπολογιστή

Ο υπολογιστής δεν μπορεί να 'κοιτάξει' το σχήμα 8.3 όπως κάναμε εμείς τώρα. Χρειάζεται μια αυτόματη μέθοδο επαλήθευσης. Με y δηλώνουμε μια μεταβλητή που αρχικά τίθεται ίση με 0. Ο υπολογιστής δημιουργεί ακολουθίες γεγονότων που επιτρέπονται από το διάγραμμα μεταβάσεων. Εάν $y = 0$ και η ακολουθία κάνει μια μετάβαση η οποία αντιστοιχεί στη λήψη της ACK0 από τον αποστολέα, τότε ο υπολογιστής θέτει $y = 1$. Εάν $y = 0$ και η μετάβαση αντιστοιχεί στη λήψη του πακέτου 1 από τον παραλήπτη, ο υπολογιστής θέτει $y = 0$.

Παρόμοια, αν $y = 1$ και η μετάβαση αντιστοιχεί στη λήψη της ACK1 από τον αποστολέα, τότε ο υπολογιστής θέτει $y = 0$. Εάν $y = 1$ και η μετάβαση αντιστοιχεί στη λήψη του πακέτου 0 από τον παραλήπτη, τότε ο υπολογιστής θέτει $y = 0$. Σημειώνουμε ότι το y φτάνει να γίνει 0 εάν και μόνο εάν η ακολουθία των καταστάσεων στο ABP αντιστοιχεί στη λήψη του πακέτου με αριθμό 0 ακολουθούμενη από την λήψη του πακέτου με αριθμό 1 χωρίς ενδιάμεση λήψη της ACK0, ή λήψη του πακέτου με αριθμό 1 ακολουθούμενη από την λήψη του πακέτου με αριθμό 0 χωρίς ενδιάμεση λήψη της ACK1.



Σχήμα 8.4 Οι μεταβάσεις της μεταβλητής y

Για την επαλήθευση του πρωτοκόλλου, θα μπορούσε να ζητήσει από τον υπολογιστή να δημιουργεί τυχαίες ακολουθίες. Εάν ο υπολογιστής δημιουργεί ακολουθίες των τιμών x και y για ένα μεγάλο χρονικό διάστημα, χωρίς το y να φτάνει ποτέ το θ , η πιθανότητα να βρεθεί μη αποδεκτή ακολουθία είναι πολύ μικρή. Μία τέτοια προσέγγιση είναι μια στοιχειώδης μέθοδος μερικής επαλήθευσης. Η μεταβλητή y καλείται παρατηρητής (monitor) της ΜΠΚ x επειδή παρακολουθεί την ακολουθία των καταστάσεων που περνάει το x για να καθορίσει εάν είναι αποδεκτή.

ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΕΦΑΡΜΟΓΗ ΤΩΝ ΜΗΧΑΝΩΝ **ΠΕΠΕΡΑΣΜΕΝΩΝ ΚΑΤΑΣΤΑΣΕΩΝ**

ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ

ΚΕΦΑΛΑΙΟ 9

Εισαγωγή

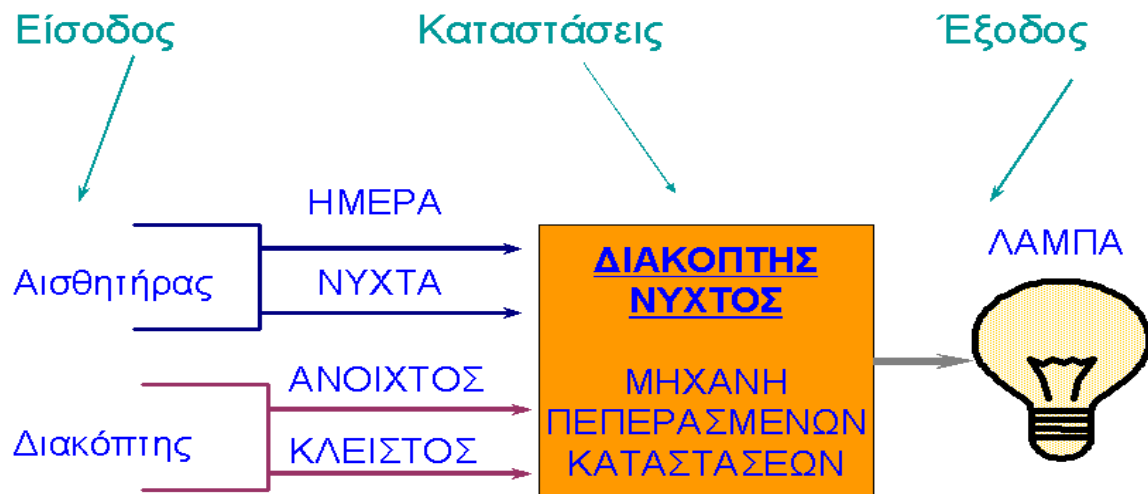
Η εφαρμογή που θα περιγράψουμε αναφέρεται στην μελέτη και σχεδίαση ενός *Διακόπτη Νυχτός* ως Μηχανή Πεπερασμένων Καταστάσεων και συγκεκριμένα μιας ασύγχρονης Mealy. Σε αυτήν την ενότητα χρησιμοποιούμε αναλυτικά τα βήματα σχεδίασης μιας ΜΠΚ, καθώς και όλους τους τρόπους αναπαράστασης (Διάγραμμα μετάβασης καταστάσεων, Πίνακας μετάβασης καταστάσεων, Διάγραμμα ροής). Στη συνέχεια, σχεδιάζουμε τον Διακόπτη Νυχτός μέσω του μικροελεγκτή BX-24 της NetMedia, ώστε να επιτελεί την λειτουργία της ΜΠΚ που θα περιγράψουμε διεξοδικά στο αρχικό στάδιο.

Η μηχανή αυτή είναι ένα ακολουθιακό σύστημα η οποία χρησιμοποιείται ευρέως και έχει εφαρμογές κυρίως σε οικιακούς χώρους, αλλά και χώρους φύλαξης. Η λειτουργικότητα του έγκειται στο γεγονός ότι η λάμπα θα ανάβει όταν νυχτώνει. Κατά τη διάρκεια της ημέρας, σχεδιάζουμε έτσι το σύστημα, ώστε να υπάρχει η δυνατότητα χειροκίνητου ελέγχου της κατάστασης της λάμπας μέσω ενός διακόπτη. Επιτελώντας τη συγκεκριμένη λειτουργία εξασφαλίζουμε σ' ένα πολύ μεγάλο βαθμό την αξιόπιστη εφαρμογή αυτού του συστήματος.

Για παράδειγμα, έστω ότι ο Διακόπτης Νυχτός έχει τοποθετηθεί στον εξωτερικό ή εσωτερικό χώρο ενός σπιτιού, όπου θέλει να διαβεί ένας άνθρωπος μεγάλης ηλικίας. Αν δεν έχει ενεργοποιηθεί αυτόματα η λάμπα (το αισθητήριο έχει ανιχνεύσει νύχτα) και υπάρχει η ανάγκη για φωτισμό του χώρου, τότε να μπορεί ο ηλικιωμένος άνθρωπος να θέτει σε λειτουργία την λάμπα μέσω του διακόπτη.

ΚΕΦΑΛΑΙΟ 10 Στάδιο 1^ο

10.1 Κατανόηση Εφαρμογής



Σχήμα 10.1 Σχεδίαση μιας εικόνας

Γνωρίζοντας πλέον την εφαρμογή που θα υλοποιήσουμε, θέτουμε τις προδιαγραφές της μηχανής μας για να περιγράψουμε την ακριβή λειτουργία του Διακόπτη Νυχτός. Δηλαδή, προσδιορίζουμε τα χαρακτηριστικά γνωρίσματα της Μηχανής Πεπερασμένων Καταστάσεων. Έτσι, δηλώνουμε τις καταστάσεις, τις εισόδους, τις λειτουργίες εξόδου και επόμενης κατάστασης καθώς και τους κανόνες μετάβασης. Αυτά τα στοιχεία καθορίζουν την επιθυμητή συμπεριφορά του ακολουθιακού μας συστήματος. Συνεπώς, αναλυτικά θα έχουμε :

Καταστάσεις

Η επιλογή πραγματοποιείται έχοντας υπόψη μας την συνολική λειτουργία του Διακόπτη Νυχτός μέσω των εισόδων (γεγονότα). Με αυτό τον τρόπο καλύπτουμε όλες τις πιθανές καταστάσεις που μπορεί να διαμορφώσει η μηχανή μας, χωρίς να υπάρχει ο κίνδυνος να μην προβλέψουμε κάποια. Εξάλλου, εάν κάποια κατάσταση που θα έχουμε ορίσει είναι περιττή, θα την παραλείψουμε κατά την εφαρμογή του αλγόριθμου απλοποίησης(ισοδύναμες καταστάσεις). Συνεπώς, προσδιορίζουμε τέσσερις πεπερασμένες καταστάσεις ως εξής :

OFF (Θεωρούμε ότι η λάμπα είναι κλειστή μέσω του διακόπτη).

ON (Θεωρούμε ότι η λάμπα είναι ανοιχτή μέσω του διακόπτη).

OFF(Sensor) (Θεωρούμε την λάμπα κλειστή μέσω του αισθητήρα).

ON(Sensor) (Θεωρούμε την λάμπα ανοιχτή μέσω του αισθητήρα).

Είσοδοι

Οι είσοδοι που λαμβάνει η μηχανή μας είναι τα γεγονότα που σε συνδυασμό με τις παρούσες καταστάσεις οδηγούν το σύστημα στην επόμενη κατάσταση. Συνεπώς, προσδιορίζουμε δύο πεπερασμένες εισόδους ως εξής:

Αισθητήρας → Η λειτουργία του είναι διπλή, παρέχει τη δυνατότητα ανίχνευσης ημέρας ή νύχτας, σύμφωνα με το φυσικό φωτισμό του χώρου.

Διακόπτης → Είναι ένας απλός διακόπτης ο οποίος μπορεί να είναι ανοιχτός ή κλειστός.

Έξοδος

Η λάμπα (είτε ανοιχτή είτε κλειστή).

Κανόνες μετάβασης

Επιθυμούμε όταν η μηχανή μας βρίσκεται στην κατάσταση OFF ή OFF (S) και ανοίξουμε τον διακόπτη, το σύστημά μας να μεταβαίνει μόνιμα στην κατάσταση ON και η λάμπα να είναι ανοιχτή. Ενώ, όταν βρίσκεται στην κατάσταση ON (S) και ανοίξουμε τον διακόπτη να μην ανταποκρίνεται σε αυτόν, αλλά μέσω του αισθητήρα σε τυχόν διαφορετική λειτουργία της μηχανής.

Στάδιο 2^ο

10.2 Μέθοδοι Αναπαράστασης

Διάγραμμα μετάβασης καταστάσεων

Από την στιγμή που έχουμε ορίσει αναλυτικά τις προδιαγραφές του Διακόπτη Νυχτός, είμαστε σε θέση να αναπαραστήσουμε τη λειτουργία που επιτελεί το σύστημα χρησιμοποιώντας αρχικά το διάγραμμα μετάβασης καταστάσεων. Σε αυτό το σημείο να επισημάνουμε ότι χρησιμοποιούμε την τεχνική σχεδίασης της μηχανής Mealy.

Ακολουθεί το διάγραμμα μετάβασης καταστάσεων και μια συνοπτική περιγραφή για το πως λειτουργεί ο Διακόπτης Νυχτός.



Σχήμα 10.2 Διάγραμμα μετάβασης καταστάσεων Διακόπτη Νυχτός

Έτσι, έχουμε τις τέσσερις καταστάσεις (OFF, ON, ON(S), OFF(S)) του Διακόπτη Νυχτός, που αναπαρίστανται μέσα στους κύκλους και δηλώνουν τις παρούσες καταστάσεις της μηχανής μας. Κάθε κατάσταση εκφράζεται από κάποιους κανόνες μετάβασης, οι οποίοι αναγράφονται πάνω στα κατευθυνόμενα τόξα. Ο κανόνας μετάβασης αποτελείται από τις εισόδους του συστήματος, δηλαδή τη λειτουργία του Διακόπτη και του Αισθητήρα, και από τις αντίστοιχες εξόδους. Να επισημάνουμε ότι σε αυτό το στάδιο της σχεδίασης χρησιμοποιούμε αλφαβητικούς χαρακτήρες, ώστε να είναι πιο κατανοητή η περιγραφή μας.

Πίνακας μετάβασης καταστάσεων / Διάγραμμα ροής

Σε αυτό το στάδιο της σχεδίασης θα περιγράψουμε αναλυτικά και διαδοχικά για κάθε κατάσταση πως η μηχανή μας ανταποκρίνεται. Αυτό το επιτυγχάνουμε με την χρησιμοποίηση των πινάκων μετάβασης καταστάσεων. Επίσης, δίπλα σε κάθε πίνακα παρουσιάζουμε και το αντίστοιχο διάγραμμα ροής.

Πίνακας I / Διάγραμμα ροής I

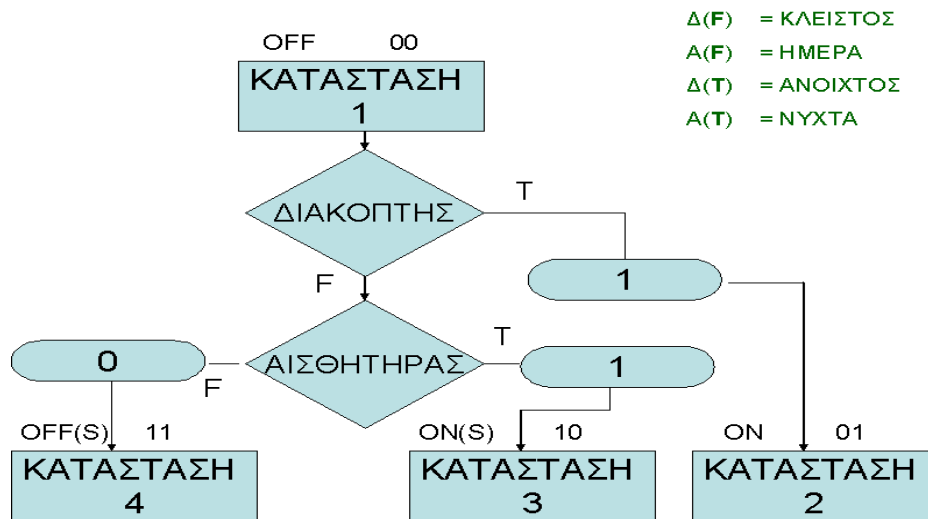
Όταν ο διακόπτης βρίσκεται στην κατάσταση **OFF** και δεχτεί ως είσοδο κλειστό διακόπτη και το αισθητήριο ανιχνεύει ημέρα, η μηχανή μεταβαίνει στην κατάσταση **OFF (S)** και η λάμπα παραμένει κλειστή. Αντίστοιχα, εάν ο διακόπτης είναι κλειστός και ο αισθητήρας ανιχνεύει νύχτα, μεταβαίνει στην κατάσταση **ON (S)** και ανοίγει η λάμπα. Στη συνέχεια, και όταν ο διακόπτης είναι ανοιχτός, ανεξάρτητα το τι λαμβάνει το αισθητήριο, έχουμε μετάβαση στην κατάσταση **ON** και η λάμπα θα παραμείνει ανοιχτή μέχρι να κλείσουμε το διακόπτη και να τεθεί σε λειτουργία ο αυτοματισμός του συστήματος.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ ΛΑΜΠΙΑ
	ΔΙΑΚ/ΤΗΣ	ΑΙΣ/ΡΑΣ		
OFF	Κλειστός	Ημέρα	OFF (S)	Κλειστή
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή
	Ανοιχτός	Ημέρα	ON	Ανοιχτή
	Ανοιχτός	Νύχτα	ON	Ανοιχτή

Πίνακας 7 Κατάσταση OFF

Όσο αφορά το διάγραμμα ροής, περιγράφουμε την ίδια λειτουργία της παρούσας κατάστασης. Στο ορθογώνιο κιβώτιο δηλώνουμε τις καταστάσεις της μηχανής, ενώ στους ρόμβους δηλώνουμε τις συνθήκες εισόδου που επικρατούν κάθε φορά. Στην προκειμένη περίπτωση τον διακόπτη και τον αισθητήρα. Οι μεταβάσεις επιβεβαιώνονται ως αληθής (True) ή ψευδής (False). Για το διακόπτη αναλογούν στο αν είναι ανοιχτός ή κλειστός αντίστοιχα, για τον

αισθητήρα αναλογούν στο αν ανιχνεύει ημέρα ή νύχτα. Τέλος, μέσα στο terminator αναγράφεται η έξοδος της μηχανής ως 0 ή 1, που αντιστοιχεί σε κλειστή ή ανοιχτή λάμπα. Η έξοδος παράγεται συναρτήσει της παρούσας κατάστασης και των εισόδων (Mealy).



Σχήμα 10.3 Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 1

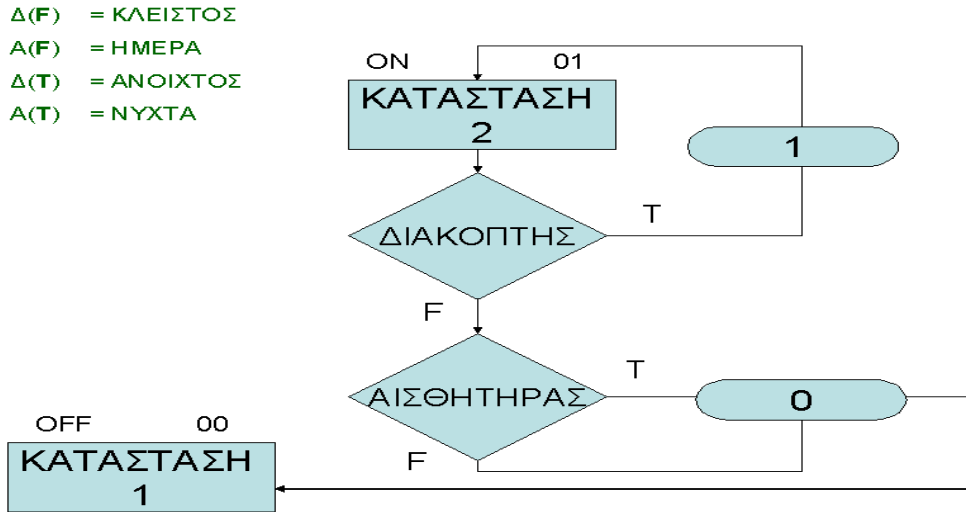
Πίνακας II / Διάγραμμα ροής II

Όταν η μηχανή μας βρίσκεται στην κατάσταση **ON** και ο διακόπτης είναι κλειστός, ανεξάρτητα το τι ανιχνεύει το αισθητήριο – ημέρα ή νύχτα -, το σύστημα μεταβαίνει στην κατάσταση **OFF** και η λάμπα κλείνει. Τώρα, από αυτή την κατάσταση “αποφασίζει” την περαιτέρω λειτουργία. Αντίθετα αν ο διακόπτης είναι ανοιχτός, είτε το αισθητήριο ανιχνεύει ημέρα είτε νύχτα παραμένει στην τρέχουσα κατάσταση και η λάμπα ανοίγει μέχρι να κλείσει ο διακόπτης. Με αυτούς τους κανόνες μετάβασης που χαρακτηρίζουν την κατάσταση **ON** του Διακόπτη Νυχτός, ο αυτοματισμός βγαίνει εκτός λειτουργίας και έτσι εξασφαλίζεται η χειροκίνητη οδήγηση του συστήματος.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ ΛΑΜΠΙΑ
	ΔΙΑΚ/ΤΗΣ	ΑΙΣ/ΡΑΣ		
ON	Κλειστός	Ημέρα	OFF	Κλειστή
	Κλειστός	Νύχτα	OFF	Κλειστή
	Ανοιχτός	Ημέρα	ON	Ανοιχτή
	Ανοιχτός	Νύχτα	ON	Ανοιχτή

Πίνακας 8 Κατάσταση ON

Στο αντίστοιχο διάγραμμα ροής έχουμε τη δυνατότητα να παρατηρήσουμε το εξής: Όταν η συνθήκη εισόδου του διακόπτη είναι False η μηχανή θα εξετάσει και τη συνθήκη εισόδου του αισθητήρα για να υπολογίσει την επόμενη κατάσταση. Ενώ όταν επαληθευτεί ως True η συνθήκη εισόδου του διακόπτη το σύστημα παραμένει στην ίδια κατάσταση.



Σχήμα 10.4 Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 2

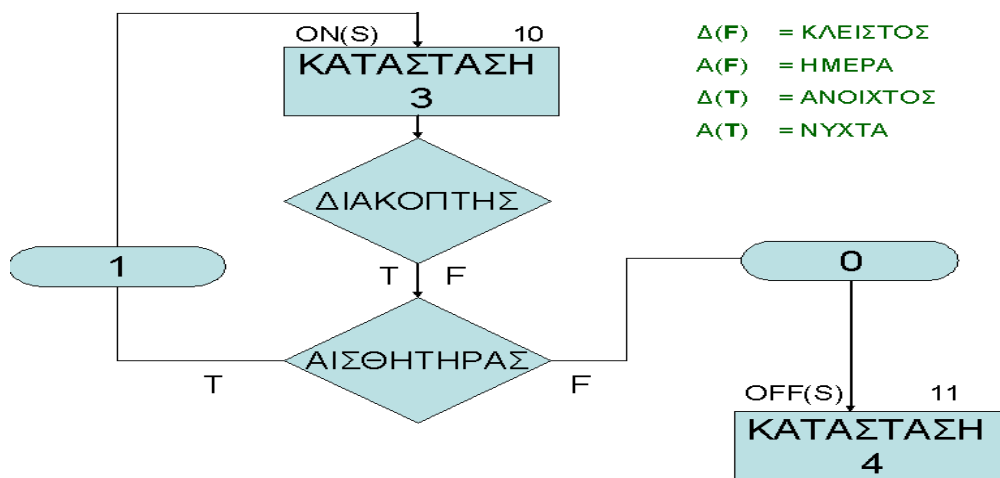
Πίνακας III / Διάγραμμα ροής III

Όταν ο Διακόπτης Νυχτός είναι στην κατάσταση **ON (S)** εκφράζεται από τους εξής κανόνες μετάβασης : Στην περίπτωση όπου ο διακόπτης είναι κλειστός και ο αισθητήρας ανιχνεύει ημέρα, τότε η επόμενη κατάσταση που μεταβαίνει είναι η **OFF (S)** και η λάμπα κλείνει. Ενώ, όταν ο αισθητήρας “βλέπει” νύχτα, η μηχανή θα παραμείνει στην παρούσα κατάσταση και η λάμπα ανοιχτή. Όμως όταν ο διακόπτης είναι ανοιχτός και έχουμε ανίχνευση ημέρας το σύστημα μεταβαίνει στην κατάσταση **OFF (S)** με τη λάμπα κλειστή, ενώ όταν ο αισθητήρας ανιχνεύει νύχτα παραμένει στην ίδια κατάσταση και η λάμπα παραμένει επίσης ανοιχτή.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ ΛΑΜΠΙΑ
	ΔΙΑΚ/ΤΗΣ	ΑΙΣ/ΡΑΣ		
ON (S)	Κλειστός	Ημέρα	OFF (S)	Κλειστή
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή
	Ανοιχτός	Ημέρα	OFF (S)	Κλειστή
	Ανοιχτός	Νύχτα	ON (S)	Ανοιχτή

Πίνακας 9 Κατάσταση ON (S)

Το διάγραμμα ροής που αναλογεί στην ίδια κατάσταση παρουσιάζεται παρακάτω. Παρατηρούμε ότι ανεξάρτητα την συνθήκη εισόδου που επικρατεί στο διακόπτη η μηχανή θα υπολογίσει την επόμενη κατάσταση εξετάζοντας την συνθήκη που θα υπάρχει στον αισθητήρα. Επισημαίνουμε ότι το σύστημα δεν αντιδρά στο άνοιγμα του διακόπτη καθώς θεωρούμε ότι όταν η λάμπα είναι ήδη ανοιχτή δεν υπάρχει λόγος να ανοίξουμε τον διακόπτη. Εκτός και αν επιθυμούμε μετά από κάποιο χρονικό διάστημα που ο αισθητήρας δεν θα ανιχνεύει νύχτα η μηχανή μας να μεταβεί στην κατάσταση OFF (S).



Σχήμα 10.4 Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 3

Πίνακας IV / Διάγραμμα ροής IV

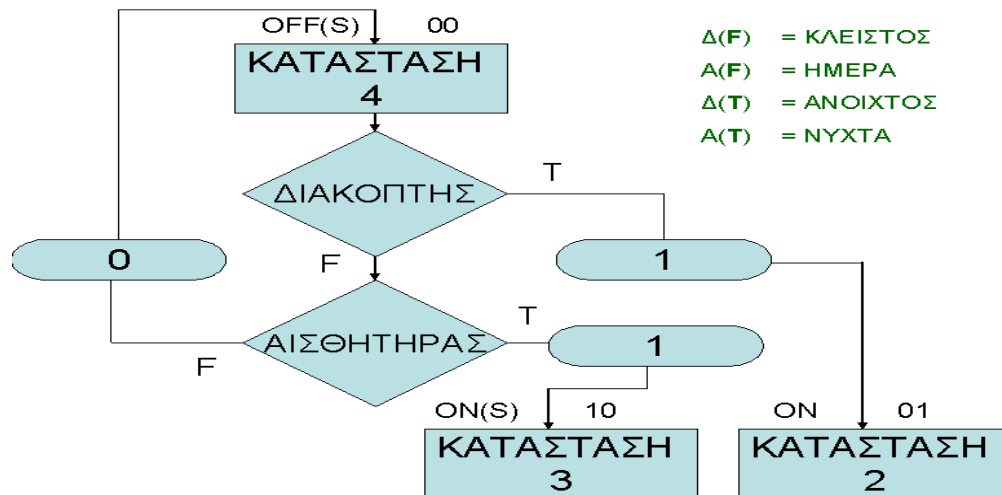
Όταν ο διακόπτης βρίσκεται στην κατάσταση OFF (S) και ο διακόπτης είναι κλειστός η μηχανή θα υπολογίσει την επόμενη κατάσταση και την αντίστοιχη έξοδο με βάση το αισθητήριο, έτσι αν ανιχνεύει ημέρα μεταβαίνει στην κατάσταση OFF (S) και η λάμπα παραμένει κλειστή. Αντίστοιχα, εάν ανιχνεύει νύχτα, μεταβαίνει στην κατάσταση ON (S) και ανοίγει η λάμπα. Στην περίπτωση που ο διακόπτης είναι ανοιχτός, ανεξάρτητα το τι λαμβάνει το αισθητήριο, έχουμε μετάβαση στην κατάσταση ON και η λάμπα θα παραμείνει ανοιχτή.

Την ίδια λειτουργία διακρίνουμε και στο διάγραμμα ροής που παρατηρούμε το σύστημα να εξετάζει και τις δύο συνθήκες εισόδων από την στιγμή που ο διακόπτης είναι False.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΞΕΟΔΟΣ ΛΑΜΠΙΑ
	ΔΙΑΚ/ΤΗΣ	ΑΙΣ/ΡΑΣ		
OFF (S)	Κλειστός	Ημέρα	OFF (S)	Κλειστή
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή
	Ανοιχτός	Ημέρα	ON	Ανοιχτή
	Ανοιχτός	Νύχτα	ON	Ανοιχτή

Πίνακας 10 Κατάσταση OFF (S)

Την ίδια λειτουργία διακρίνουμε και στο διάγραμμα ροής που παρατηρούμε το σύστημα να εξετάζει και τις δύο συνθήκες εισόδων από την στιγμή που ο διακόπτης είναι False.



Σχήμα 10.5 Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 4

Στάδιο 3^ο

10.3 Ελαχιστοποίηση καταστάσεων

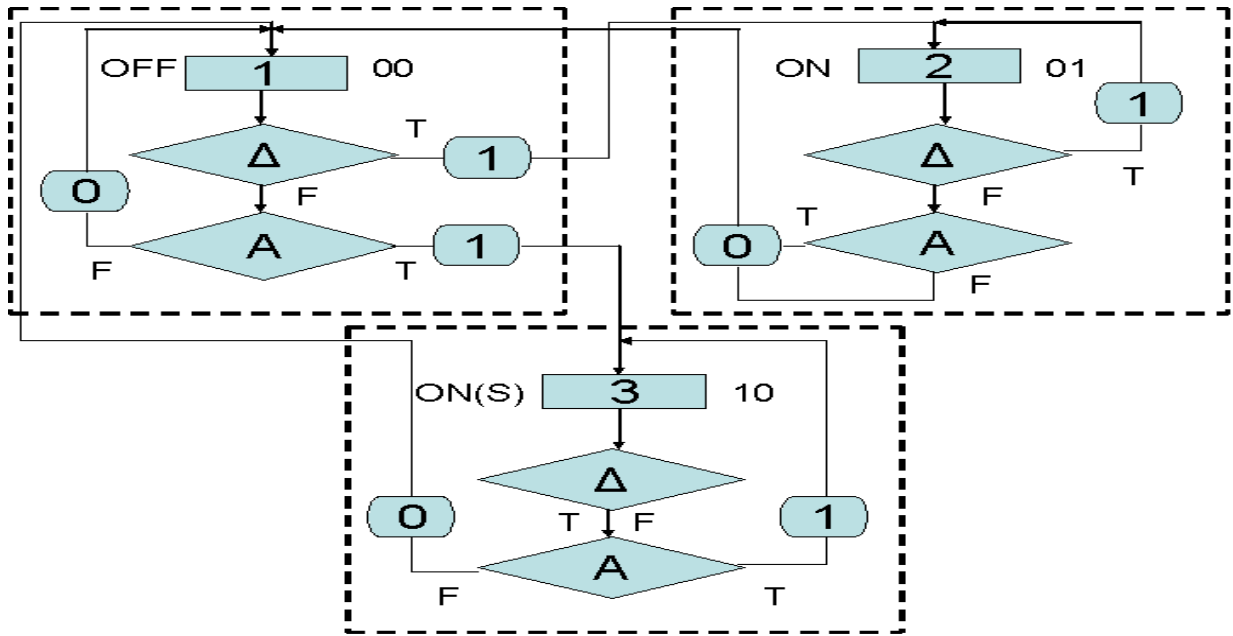
Τέλος, παραθέτουμε την τελική μορφή του πίνακα μετάβασης καταστάσεων που περιγράψαμε αναλυτικά στις προηγούμενες παραγράφους. Έχουμε τη δυνατότητα παρατηρώντας στη τελική μορφή τον πίνακα να διαγράψουμε τυχόν ισοδύναμες καταστάσεις, χρησιμοποιώντας τον αλγόριθμο απλοποίησης. Αυτό θα μας βοηθήσει και στην απλοποίηση της υλοποίησης του hardware του Διακόπτη Νυχτός γιατί θα έχουμε

λιγότερες καταστάσεις από αυτές που είχαμε ορίσει στο αρχικό στάδιο μελέτης της συγκεκριμένης Μηχανής Πεπερασμένων Καταστάσεων.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ	ΕΞΟΔΟΣ (ΛΑΜΠΙΑ)
	ΔΙΑΚΟΠΤΗΣ	ΑΙΣΘΗΤΗΡΑΣ		
OFF	Κλειστός	Ημέρα	OFF (S) (OFF)	Κλειστή
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή
	Ανοιχτός	Ημέρα	ON	Ανοιχτή
	Ανοιχτός	Νύχτα	ON	Ανοιχτή
ON	Κλειστός	Ημέρα	OFF	Κλειστή
	Κλειστός	Νύχτα	OFF	Κλειστή
	Ανοιχτός	Ημέρα	ON	Ανοιχτή
	Ανοιχτός	Νύχτα	ON	Ανοιχτή
ON (S)	Κλειστός	Ημέρα	OFF (S) (OFF)	Κλειστή
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή
	Ανοιχτός	Ημέρα	OFF (S) (OFF)	Κλειστή
	Ανοιχτός	Νύχτα	ON (S)	Ανοιχτή
OFF (S)	Κλειστός	Ημέρα	OFF (S)	Κλειστή X
	Κλειστός	Νύχτα	ON (S)	Ανοιχτή X
	Ανοιχτός	Ημέρα	ON	Ανοιχτή X
	Ανοιχτός	Νύχτα	ON	Ανοιχτή X

Πίνακας 11 Συνολικός Πίνακας

Επομένως, διακρίνουμε ότι η κατάσταση **OFF** και η κατάσταση **OFF (S)** είναι ισοδύναμες. Δηλαδή, εξετάζοντας τους κανόνες μετάβασης των δύο συγκεκριμένων καταστάσεων, διαπιστώνουμε ότι για τον ίδιο συνδυασμό εισόδων παράγουν την ίδια έξοδο και μεταβαίνουν στην ίδια επόμενη κατάσταση. Άρα, σύμφωνα με τον αλγόριθμο απλοποίησης διαγράφουμε την κατάσταση OFF (S) και την αντικαθιστούμε με την ισοδύναμή της – OFF – στον υπόλοιπο πίνακα μετάβασης. Επίσης, παρουσιάζουμε και σε ενιαία μορφή το διάγραμμα ροής, που αποτελείται πλέον από τρία μπλοκ που περιγράφουν τις 3 καταστάσεις της μηχανής.



Σχήμα 10.6 Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 5

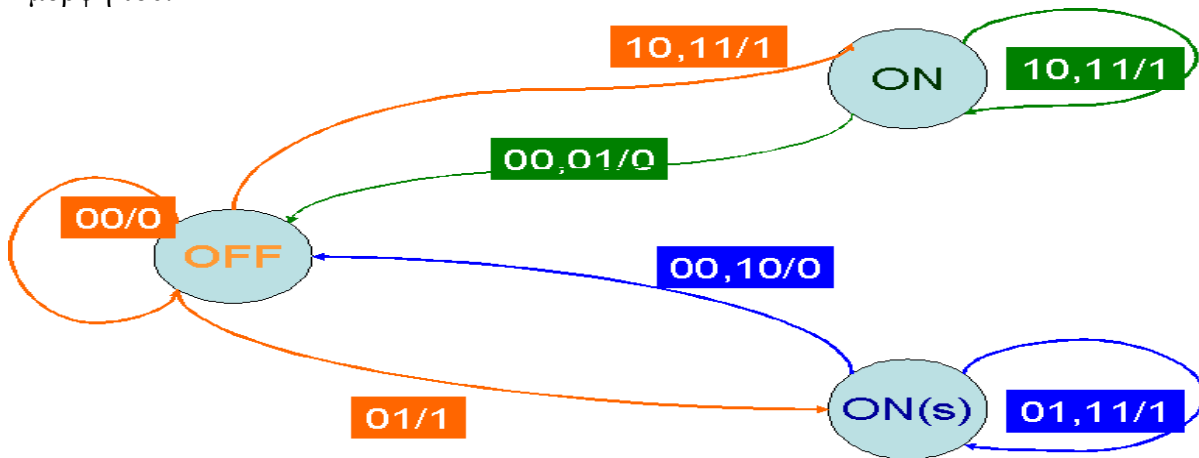
Στάδιο 4^ο

10.4 Κωδικοποίηση Δεδομένων

Μετά και την απλοποίηση των καταστάσεων, μπορούμε να αρχίσουμε την κωδικοποίηση των δεδομένων του Διακόπτη Νυχτός, ώστε σταδιακά να υλοποιήσουμε το ακολουθιακό μας κύκλωμα. Επιλέγουμε το φυσικό δυαδικό κώδικα (Binary Code) για να κωδικοποιήσουμε τις καταστάσεις, τις εισόδους και την έξοδο. Αναλυτικότερα:

- **Καταστάσεις** Off = 00 , ON = 01, ON (S) = 10
- **Είσοδοι** Διακόπτης κλειστός / ανοιχτός = 0 / 1. Ανίχνευση ημέρας / νύχτας = 0 / 1
- **Έξοδος** Λάμπα κλειστή / ανοιχτή = 0 / 1

Επίσης, ακολουθεί και το διάγραμμα μετάβασης καταστάσεων στην κωδικοποιημένη μορφή του.



Σχήμα 10.7 Κωδικοποιημένο Διάγραμμα μετάβασης καταστάσεων

Από την στιγμή που έχουμε 3 καταστάσεις, θα χρησιμοποιήσουμε δύο δυφία στην κωδικοποίηση των καταστάσεων. Αυτό συνεπάγεται με τη χρησιμοποίηση δύο φλιπ – φλοπ τύπου D (Delay F-F) για να υλοποιήσουμε τον καταχωρητή καταστάσεων του συστήματός μας. Το πλήθος των φ-φ εξαρτάται από την σχέση . Η επιλογή του συγκεκριμένου τύπου φ-φ δεν εξαρτάται από κάποιο κανόνα ή προϋπόθεση. Έτσι, έχουμε τον κωδικοποιημένο πίνακα μετάβασης καταστάσεων.

ΠΑΡΟΥΣΑ ΚΑΤΑΣΤΑΣΗ		ΕΙΣΟΔΟΙ		ΕΠΟΜΕΝΗ ΚΑΤΑΣΤΑΣΗ		ΕΞΟΔΟΣ (ΛΑΜΠΙΑ)
SA	SB	ΔΙΑΚΟΠΤΗΣ	ΑΙΣΘΗΤΗΡΑΣ	DA	DB	
0	0	0	0	0	0	0
0	0	0	1	1	0	1
0	0	1	0	0	1	1
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	0	1	0	0	0
0	1	1	0	0	1	1
0	1	1	1	0	1	1
1	0	0	0	0	0	0
1	0	0	1	1	0	1
1	0	1	0	0	0	0
1	0	1	1	1	0	1

Πίνακας 12 Κωδικοποιημένος Πίνακας
Στάδιο 5^ο

10.5 Επιλογή τύπου φλιπ - φλοπ

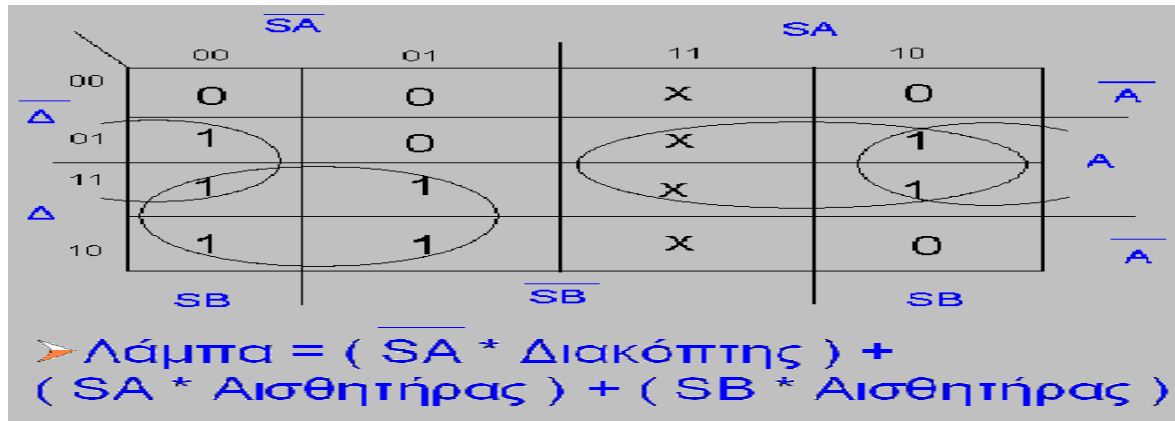
Στην στήλη της επόμενης κατάστασης προσδιορίζουμε τις εισόδους DA και DB των δύο φ-φ που θα χρησιμοποιήσουμε. Στη συνέχεια με τη βοήθεια των πινάκων Καρνώ θα εξάγουμε τις λογικές συναρτήσεις του συνδυαστικού κυκλώματος που υπολογίζει την επόμενη κατάσταση της μηχανής καθώς επίσης και του συνδυαστικού κυκλώματος που υπολογίζει την έξοδο του συστήματος. Εφόσον έχουμε δύο εισόδους και εκφράζουμε τις παρούσες καταστάσεις με δύο δυφία ο πίνακας Καρνώ θα αποτελείται από 16 τετράγωνα.

$DA = (SA * Αισθητήρας) + (SB * Διακόπτης * Αισθητήρας)$
 $DB = (\overline{SA} * Διακόπτης)$

The image shows two Karnaugh maps. The first map is for DA, with SA on the horizontal axis (00, 01, 11, 10) and SB on the vertical axis (00, 01, 11, 10). The output values are 0, 0, X, 0 for SA=00; 1, 0, X, 1 for SA=01; 0, 0, X, 1 for SA=11; 0, 0, X, 0 for SA=10. Circled 1s are at (01,0) and (11,1). The second map is for DB, with SA on the horizontal axis (00, 01, 11, 10) and SB on the vertical axis (00, 01, 11, 10). The output values are 0, 0, X, 0 for SA=00; 0, 0, X, 0 for SA=01; 1, 1, X, 0 for SA=11; 1, 1, X, 0 for SA=10. Circled 1s are at (11,0) and (10,0).

Σχήμα 10.8 Πίνακας Καρνώ 1

Οι ομαδοποιήσεις μας εξάγουν τις λογικές Boolean εκφράσεις των συνδυαστικών κυκλωμάτων. Αυτές ισοδυναμούν με την ορθή και ανάστροφη λειτουργία του διακόπτη, του αισθητήρα και των δύο παρουσιών καταστάσεων.

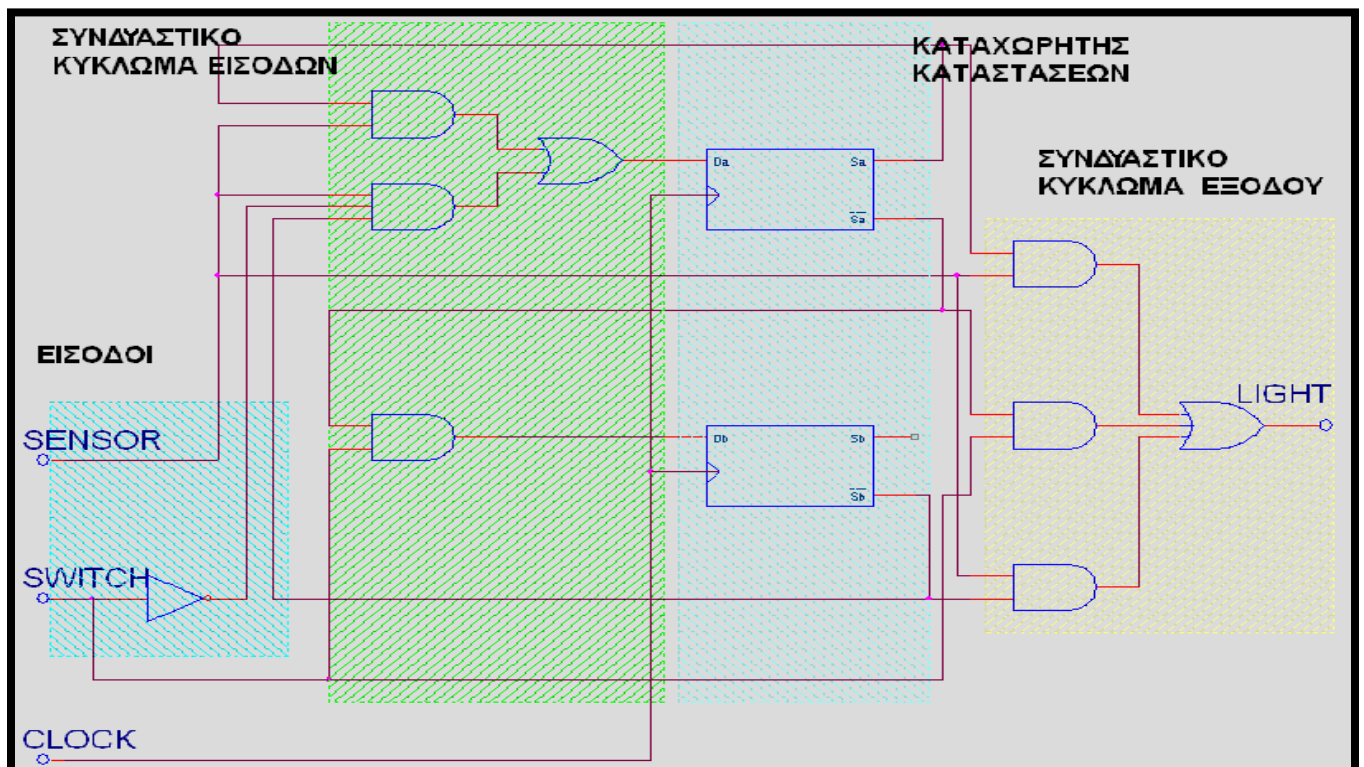


Σχήμα 10.9 Πίνακας Καρνώ 2

Στάδιο 6^ο

10.6 Τελική υλοποίηση του ακολουθιακού κυκλώματος

Έχουμε φτάσει στο τελικό στάδιο της σχεδίασης μιας Μηχανής Πεπερασμένων Καταστάσεων και παρουσιάζουμε την υλοποίηση του Διακόπτη Νυχτός (Ασύγχρονη Mealy)

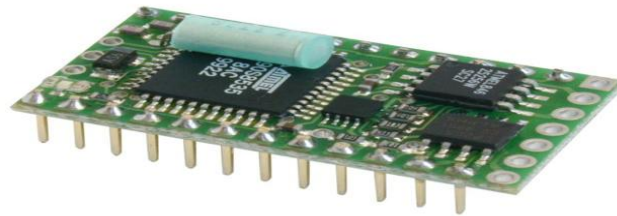


Σχήμα 10.10 Τελικό κύκλωμα

. Διακρίνουμε τα δύο μπλοκ της συνδυαστικής λογικής που αποτελούνται από 9 λογικές πύλες (1 πύλη NOT, 6 πύλες AND και 2 πύλες OR). Η μνήμη του συστήματος που αποθηκεύει τις καταστάσεις της μηχανής αποτελείται από τα 2 φ-φ τύπου D. Επίσης, διακρίνουμε επιπλέον μια βαθμίδα που καθορίζει τις συσκευές των εισόδων. Οι τέσσερις αυτές βαθμίδες σε σκιαγραφημένο φόντο χαρακτηρίζουν τον τύπο της ΜΠΚ. Οι έξοδοι των φ-φ που είναι οι επόμενες καταστάσεις της Mealy ανατροφοδοτούνται και δίνουν την παρούσα κατάσταση. Τέλος, η έξοδος είναι λειτουργία τόσο της παρούσας κατάστασης όσο και των εισόδων του κυκλώματος. Λογικό, γιατί πρόκειται για μηχανή Mealy.

ΚΕΦΑΛΑΙΟ 11

Λειτουργία του Διακόπτη Νυχτός με τον BX-24



Σχήμα 11.1 BX-24

11.1 BasicX

Ο BasicX είναι ένα πλήρες σύστημα ελέγχου ενσωματωμένο σε ένα ολοκληρωμένο κύκλωμα το οποίο σε συνδυασμό με το λογισμικό του μπορεί να τρέξει εφαρμογές σε περιβάλλον Windows. Ο συνδυασμός του ολοκληρωμένου κυκλώματος BasicX με τις πρόσθετες συσκευές, τις οποίες θα δούμε παρακάτω, καθιστούν τον BX-24 έναν αυθύπαρκτο υπολογιστή. Αυτές είναι οι εξής:

BX-24 Hardware: Το σύστημα του BX-24 περιλαμβάνει έναν γρήγορο επεξεργαστή όπου το σύστημα λειτουργίας του BasicX αποθηκεύεται στη μνήμη ROM. Επίσης περιλαμβάνει 400 bytes RAM, 32 Kbytes EEPROM και πολλές εισόδους και εξόδους όπως είναι οι χρονοστές.

BasicX Operating System(BOS): Το παραπάνω σύστημα μας παρέχει το γνωστό μας multitasking περιβάλλον, το οποίο κάνει το ολοκληρωμένο κύκλωμα του BasicX πολύ δυνατό και χρήσιμο.

BasicX Development Enviroment: Το περιβάλλον ανάπτυξης του BasicX εκτός των άλλων περιέχει και τον compiler, του οποίου την λειτουργία θα δούμε στη συνέχεια για να κατανοήσουμε καλύτερα τη χρησιμότητα και την λειτουργικότητα του.

Ο Compiler του BasicX

Αφού δημιουργήσουμε και συντάξουμε το πρόγραμμα το οποίο θέλουμε να θέσουμε σε εφαρμογή σειρά παίρνει ο compiler. Ο compiler μετατρέπει ή καλύτερα μεταφράζει

τον κώδικα του BasicX που εμείς καταλαβαίνουμε σε δυαδική μορφή, την οποία εμείς αδυνατούμε να κατανοήσουμε. Δεν πρέπει να ξεχνάμε ότι ο μικροεπεξεργαστής κατανοεί μόνο τη δυαδική μορφή και ότι όλες οι γλώσσες προγραμματισμού που έχουν επινοηθεί δεν είναι παρά ένας τρόπος για να κατανοούμε εύκολα και γρήγορα το δυαδικό σύστημα. Στη συνέχεια τα δεδομένα γράφονται σε ένα φάκελο (*.BXB). Ο compiler επίσης αναλαμβάνει τους ακροδέκτες εισόδου και εξόδου, τη ρύθμιση της μνήμης RAM καθώς και άλλες σημαντικές παραμέτρους και όλα αυτά τα στοιχεία τα τοποθετεί στο φάκελο προτιμήσεων (*.PRF). Εφόσον υπάρχουν και τα δύο είδη φακέλων η παρουσίαση του προγράμματος είναι έτοιμη και μπορεί στη συνέχεια να αποθηκευτεί σε οποιοδήποτε μέσο αποθήκευσης.

Όταν φορτώνουμε το πρόγραμμα στη μνήμη EEPROM το ολοκληρωμένο κύκλωμα του BasicX ξεκινά να εκτελεί τις οδηγίες που δέχεται από τη μνήμη EEPROM. Σε περίπτωση διακοπής του ρεύματος ο κώδικας του προγράμματος παραμένει αποθηκευμένος στη μνήμη χωρίς να χαθεί.

Συνοψίζοντας όλα τα παραπάνω εύκολα κατανοούμε ότι η συγκεκριμένη γλώσσα που χρησιμοποιούμε μας δίνει τη δυνατότητα και την ευχέρεια να δημιουργούμε σωστά δομημένα και συγκροτημένα προγράμματα σε απλή και ευκολοκατανόητη γλώσσα προγραμματισμού.

O BX-24

Ο BX-24 (σχήμα) απαιτεί συνεχή τάση που κυμαίνεται μεταξύ των 5.5V και των 15V, κάτι το οποίο τον κάνει ιδανικό για τροφοδοσία από μπαταρία. Πάνω στην πλακέτα (σχήμα) του BX-24 υπάρχουν τα εξής:

O επεξεργαστής BasicX

Ο παραπάνω επεξεργαστής είναι η καρδιά του BX-24. Εδώ διαβάζεται και εκτελείται το πρόγραμμα που είναι αποθηκευμένο στην μνήμη EEPROM. Ο BX-24 έχει 16 ακροδέκτες E/E, οι οποίοι μπορούν και να χρησιμεύσουν σαν Α/Ψ μετατροπείς για ανίχνευση και μέτρηση αναλογικών τάσεων.

SPI EEPROM chip

Εδώ αποθηκεύεται το πρόγραμμα, το οποίο εμείς γράφουμε.

Σειριακή θύρα

Μία υψηλής ταχύτητας των 5V σειριακή θύρα μας παρέχει τη δυνατότητα για επικοινωνία με το εξωτερικό περιβάλλον και άλλα περιφερειακά. Ένα επίσης πολύ σημαντικό πλεονέκτημα είναι και η αμφίδρομη επικοινωνία που υπάρχει και η οποία ανεβάζει πολύ τα στάνταρ.

Ρυθμιστής τάσης

Ο ρυθμιστής τάσης μας εφοδιάζει με τάση εισόδου, η οποία κυμαίνεται μεταξύ των 5.7 και 15V

A/Ψ μετατροπέας

Ο BX-24 περιλαμβάνει έναν 8 καναλιών των 10-bit Α/Ψ μετατροπέα. Όλα τα κανάλια μπορούν να χρησιμοποιηθούν είτε σαν αναλογικές είτε σαν ψηφιακές εισόδους. Οι παραπάνω εισοδοί δεν μπορούν να αντέξουν μεγαλύτερη ή μικρότερη τάση πέρα από το φάσμα των 0-5V.

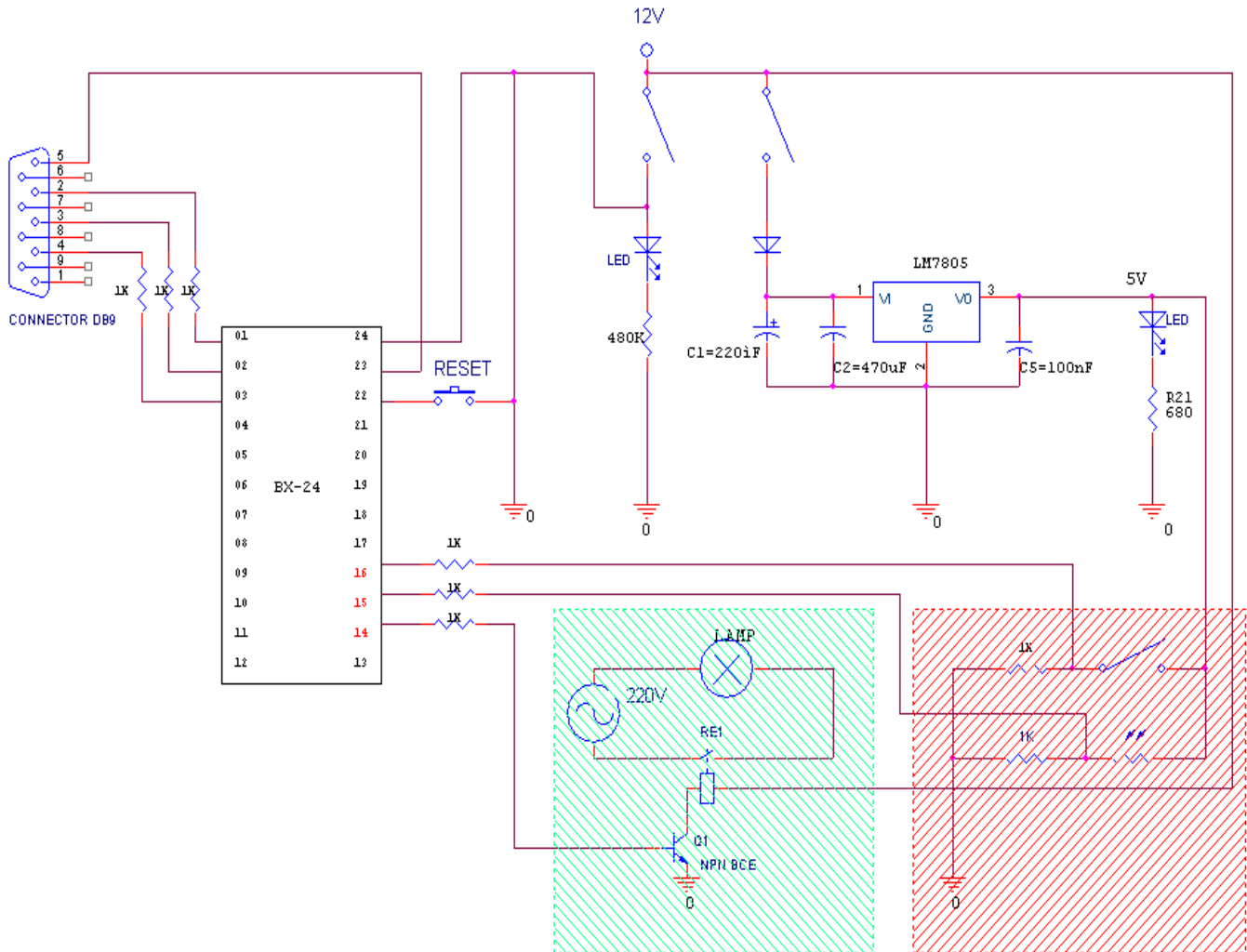
Μνήμη

Αυτές είναι η RAM και EEPROM, όπου στην τελευταία αποθηκεύεται και ο κωδικός προγράμματος. Τέλος, υπάρχουν και οι καταχωρητές χρόνου και ημερομηνίας που συμβαδίζουν με το δικό μας ημερολογιακό εικοσιτετράωρο.

Του δίνουμε τροφοδοσία από τους ακροδέκτες 23 (γείωση) και 24 και μπορούμε να του δώσουμε από 6 έως 15 volt. Στον ακροδέκτη 22 όπως βλέπουμε και από το σχήμα συνδέουμε ένα διακόπτη επαναφοράς (Reset Button) ο οποίος δίνει σήμα διακοπής, όταν πατηθεί, στον BX-24, αφού είναι συνδεδεμένος με τη γείωση. Η επικοινωνία μας με τον BX-24 γίνεται μέσω υπολογιστή με τη σειριακή θύρα, χρησιμοποιώντας τους ακροδέκτες 1,2,3,4 (αυτά φαίνονται και στο κύκλωμα του σχήματος). Οι ακροδέκτες 13-20 είναι αναλογικές εισόδους του μικροεπεξεργαστή. Δηλαδή εκεί συνδέουμε τα αισθητήρια μας. Ο BX-24 αναλαμβάνει μέσω μετατροπέων από αναλογικό σε ψηφιακό να μεταφράσει το σήμα που δέχεται. Αντίστοιχα στους ακροδέκτες 5-20 τοποθετούμε τις ψηφιακές εισόδους. Πρέπει να αναφέρουμε ότι η σύνδεση των διαφόρων εξαρτημάτων που χρησιμοποιούμε χρειάζονται κάποια βοηθητικά κυκλώματα για να μπορέσουν να συνεργαστούν με τον BX-24.

11.2 Κύκλωμα εφαρμογής

Το κύκλωμα που παρουσιάζουμε στο σχήμα αποτελεί το hardware της εφαρμογής μας. Τα βασικά του χαρακτηριστικά είναι ότι τροφοδοτούμε τον BX-24 με 12 Volt στο pin 24, και χρησιμοποιώντας έναν σταθεροποιητή τάσης (LM7805) των 5 Volt τροφοδοτούμε την φωτοαντίστασή μας και τον διακόπτη (χρησιμοποιούμε led για την ένδειξη των τάσεων). Παρατηρούμε στο κύκλωμά μας δύο σκιαγραφημένες περιοχές. Συγκεκριμένα και όσο αφορά την κόκκινη σκιαγραφημένη περιοχή διακρίνουμε τις εισόδους μας, δηλαδή τον διακόπτη και την φωτοαντίστασή μας (λειτουργεί ως αισθητήρας φωτός), που συνδέονται με τις αναλογικές εισόδους 15 και 16 του BX-24 μέσω προσαρμοστικών κυκλωμάτων. Στην πράσινη σκιαγραφημένη περιοχή παρουσιάζεται η έξοδος, δηλαδή η λάμπα, που ενεργοποιείται μέσω ενός ρελέ όταν έρθει σήμα από τον BX-24 και συγκεκριμένα από το pin 14.



Σχήμα 11.2 Κύκλωμα εφαρμογής Διακόπτη Νυχτός

11.3 Δομή προγράμματος

Στο πρόγραμμα που ακολουθεί αναθέτουμε τον διακόπτη ως d , τον αισθητήρα ως f και την πεπερασμένη κατάσταση της μηχανής ως x . Στην συνέχεια θέτουμε στην μεταβλητή x τρεις τιμές, συγκεκριμένα όπου $x = 0$ δηλώνουμε την κατάσταση off, όπου $x=1$ θεωρούμε την κατάσταση on και τέλος όπου $x=2$ την κατάσταση ons. Επειδή θέλουμε να υλοποιήσουμε μια Μηχανή Πεπερασμένων Καταστάσεων τύπου Mealy, στις συνθήκες μας γίνεται έλεγχος της τρέχουσας κατάστασης της μηχανής, αλλά και των εισόδων μας. Με αυτό τον τρόπο σε κάθε υπορουτίνα που καλούμε (a , b , c) πραγματοποιούμε την έξοδο των δύο συνδυαστικών κυκλωμάτων της μηχανής Me , δηλαδή την επόμενη κατάσταση και το άνοιγμα ή σβήσιμο της λάμπας. Υποθέτουμε ότι όπου 350, 300 είναι οι πειραματικές τιμές για τον έλεγχο των εισόδων μας.

Αρχικοποίηση

Dim d as integer *'δήλωση της μεταβλητής d ως ακέραιο'*

Dim f as integer *'δήλωση της μεταβλητής f ως ακέραιο'*

Dim x as integer *'δήλωση της μεταβλητής x ως ακέραιο'*

Κυρίως πρόγραμμα

Sub Main()

x=0 *'θέτουμε την αρχική κατάσταση off πριν αρχίσει το loop'*

do *'κάνε'*

d=getadc(15) *'δηλώνουμε ότι στο pin15 στέλνουμε αναλογικό σήμα από τον διακόπτη'*

f=getadc(16) *'δηλώνουμε ότι στο pin16 στέλνουμε αναλογικό σήμα από τον αισθητήρα'*

Κατάσταση off

if (x=0) and (d<350) and (f<300) then *'εάν είναι στην κατάσταση off και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει ημέρα'*

call a() *'κάλεσε την υπορουτίνα a'*

end if *'τέλος συνθήκης'*

if (x=0) and (d<350) and (f>300) then *'εάν είναι στην κατάσταση off και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει νύχτα'*

call c() *'κάλεσε την υπορουτίνα c'*

end if *'τέλος συνθήκης'*

if (x=0) and (d>350) and (f<300) then *'εάν είναι στην κατάσταση off και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει ημέρα'*

call b() *'κάλεσε την υπορουτίνα b'*

end if *'τέλος συνθήκης'*

if (x=0) and (d>350) and (f>300) then *'εάν είναι στην κατάσταση off και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει νύχτα'*

call b() *'κάλεσε την υπορουτίνα b'*

end if *'τέλος συνθήκης'*

Κατάσταση on

if (x=1) and (d<350) and (f<300) then *'εάν είναι στην κατάσταση on και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει ημέρα'*

call a() *'κάλεσε την υπορουτίνα a'*

end if *'τέλος συνθήκης'*

if (x=1) and (d<350) and (f>300) then *'εάν είναι στην κατάσταση on και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει νύχτα'*

call a() *'κάλεσε την υπορουτίνα a'*

end if *'τέλος συνθήκης'*

if (x=1) and (d>350) and (f<300) then *‘εάν είναι στην κατάσταση on και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει ημέρα’*
 call b() *‘κάλεσε την υπορουτίνα b’*
end if *‘τέλος συνθήκης’*

if (x=1) and (d>350) and (f>300) then *‘εάν είναι στην κατάσταση on και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει νύχτα’*
 call b() *‘κάλεσε την υπορουτίνα b’*
end if *‘τέλος συνθήκης’*

Κατάσταση ons

if (x=2) and (d<350) and (f<300) then *‘εάν είναι στην κατάσταση ons και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει ημέρα’*
 call a() *‘κάλεσε την υπορουτίνα a’*
end if *‘τέλος συνθήκης’*

if (x=2) and (d<350) and (f>300) then *‘εάν είναι στην κατάσταση ons και ο διακόπτης είναι κλειστός και η φωτοαντίσταση ανιχνεύει νύχτα’*
 call c() *‘κάλεσε την υπορουτίνα c’*
end if *‘τέλος συνθήκης’*

if (x=2) and (d>350) and (f<300) then *‘εάν είναι στην κατάσταση ons και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει ημέρα’*
 call a() *‘κάλεσε την υπορουτίνα a’*
end if *‘τέλος συνθήκης’*

if (x=2) and (d>350) and (f>300) then *‘εάν είναι στην κατάσταση ons και ο διακόπτης είναι ανοιχτός και η φωτοαντίσταση ανιχνεύει νύχτα’*
 call c() *‘κάλεσε την υπορουτίνα c’*
end if *‘τέλος συνθήκης’*

loop
End Sub

Δήλωση υπορουτινών

sub a() *‘υπορουτίνα a’*
x=0 *‘θέτει την μηχανή σε κατάσταση off’*
call putpin(14,0) *‘σβήσε τη λάμπα’*
end sub *‘τέλος υπορουτίνας’*

sub b() *‘υπορουτίνα b’*
x=1 *‘θέτει την μηχανή σε κατάσταση off’*
call putpin(14,1) *‘άνοιξε τη λάμπα’*

```
end sub          'τέλος υπορουτίνας'  
  
sub c()         'υπορουτίνα c'  
x=2            'θέτει την μηχανή σε κατάσταση off'  
call putpin(14,1) 'άνοιξε τη λάμπα'  
end sub        'τέλος υπορουτίνας'
```

ΕΠΙΛΟΓΟΣ

Οι μηχανές πεπερασμένων καταστάσεων είναι μια τεχνική σχεδίασης την οποία μπορείς να χρησιμοποιήσεις για οποιαδήποτε εφαρμογή. Σου δίνει τη δυνατότητα να οργανώνεις τη σκέψη σου και τη δομή της λύσης του προβλήματος σου. Μέσω αυτής της τεχνικής είναι δυνατόν να δημιουργήσεις και να εκτελέσεις ένα πρόγραμμα με μικρά ποσοστά αποτυχίας. Βασικός παράγοντα σε αυτό είναι να εκτελέσεις σωστά και με επιμέλεια τα βήματα της σχεδίασης, ειδικά τον καθορισμό των καταστάσεων.

ΕΛΛΗΝΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

- Γερ. Π. Φραγκάκη, Κεφάλαια 6, 7, 8 από “Λογικά Κυκλώματα”, 1975
- Νίκος Καλοκάσης, ‘Εφαρμογές ψηφιακής ακολουθιακής λογικής’, 1999
- Jean Warland, “Δίκτυα Επικοινωνιών”, 1997

ΞΕΝΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

- Leach / Malvino , “Digital Principles and Applications”, 1997
- Aggramal S, Walper P, “Adding liveness properties to coupled FSM”, 1990
- Hopcroft, J.E and J.D. Ullman, “ Introduction to automata theory, languages and computation ”, 1979

ΙΣΤΙΟΣΕΛΙΔΕΣ

- <http://www.cs.njit.edu/~elsa/775-spring2004/775%20Lecture%2025.ppt>
- <http://ocw.mit.edu/.../6-004Computation-StructuresFall2002/5DD18A5E-4E7E-4F0E-A601-5F84B5F839DE/0/L07FSMs.pdf>
- <http://www.unc.edu/courses/2004spring/comp/120/001/handouts/L06-SyncLogic.pdf>
- <http://www.cs.umd.edu/class/fall2003/cmssc311/Lectures/lecture28/fsm.pdf>
- <http://www.ecs.umass.edu/ece/tessier/courses/221/lecture/lect25-engin112.pdf>
- <http://www.spsu.edu/cs/faculty/bbrown/papers/statemachines.pdf>
- http://www.ece.osu.edu/~cglee/EE561/Lecture/Lecture11_13_2on1.pdf

ΠΑΡΑΡΤΗΜΑ Α

Κατάλογος Σημάτων

2.1	Ακολουθιακό Κύκλωμα ή Finite Automata	7
2.2	Χρονισμός φλιπ – φλοπ	8
3.1	Μπλοκ Διάγραμμα μιας απλής Μηχανής Πεπερασμένων Καταστάσεων.....	9
3.2	Μηχανή Πεπερασμένων Καταστάσεων με ξεχωριστούς αποκωδικοποιητές.....	10
3.3	Λειτουργία μνήμης ROM.....	10
4.1	Διάγραμμα καταστάσεων 1.....	11
4.2	Διάγραμμα καταστάσεων 2.....	11
4.3	Διάγραμμα καταστάσεων για μηχανές Mo και Me.....	12
4.4	Διάγραμμα καταστάσεων 3.....	12
4.5	Διάγραμμα ροής 1.....	15
4.6	Διάγραμμα ροής 2.....	15
5.1	Μηχανή Moore.....	16
5.2	Μηχανή Mealy.....	16
5.3	Ασύγχρονη μηχανή Moore.....	17
5.4	Σύγχρονη μηχανή Moore.....	17
5.5	Ασύγχρονη μηχανή Mealy.....	17
5.6	Σύγχρονη μηχανή Mealy.....	18
5.7	Διάγραμμα και Πίνακας μετάβασης καταστάσεων μηχανής Mealy.....	18
5.8	Διάγραμμα και Πίνακας μετάβασης καταστάσεων μηχανής Moore.....	19
7.1	Διάγραμμα μετάβασης καταστάσεων (Me/Mo) μηχανής πώλησης.....	22
7.2	Εικόνα ελεγκτή φωτός κυκλοφορίας.....	23
8.1	Διάγραμμα μεταβολής των καταστάσεων για καταχωρητή τεσσάρων πακέτων.....	24
8.2	Διάγραμμα μετάβασης καταστάσεων για μη αριθμημένα πακέτα.....	26
8.3	Διάγραμμα μεταβολής καταστάσεων του ABR.....	27
8.4	Οι μεταβάσεις της μεταβλητής y.....	28
10.1	Σχεδίαση μιας εικόνας.....	30
10.2	Διάγραμμα μετάβασης καταστάσεων Διακόπτη Νυχτός.....	32
10.3	Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 1.....	33
10.4	Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 2.....	34
10.4	Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 3.....	34
10.5	Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 4.....	36
10.6	Διάγραμμα ροής καταστάσεων Διακόπτη Νυχτός 5.....	38
10.7	Κωδικοποιημένο Διάγραμμα μετάβασης καταστάσεων.....	38
10.8	Πίνακας Καρνώ 1.....	39
10.9	Πίνακας Καρνώ 2.....	40
10.10	Τελικό κύκλωμα.....	40
11.1	BX-24	41
11.2	Κύκλωμα εφαρμογής Διακόπτη Νυχτός	44.

ΠΑΡΑΡΤΗΜΑ Β

Κατάλογος Πινάκων

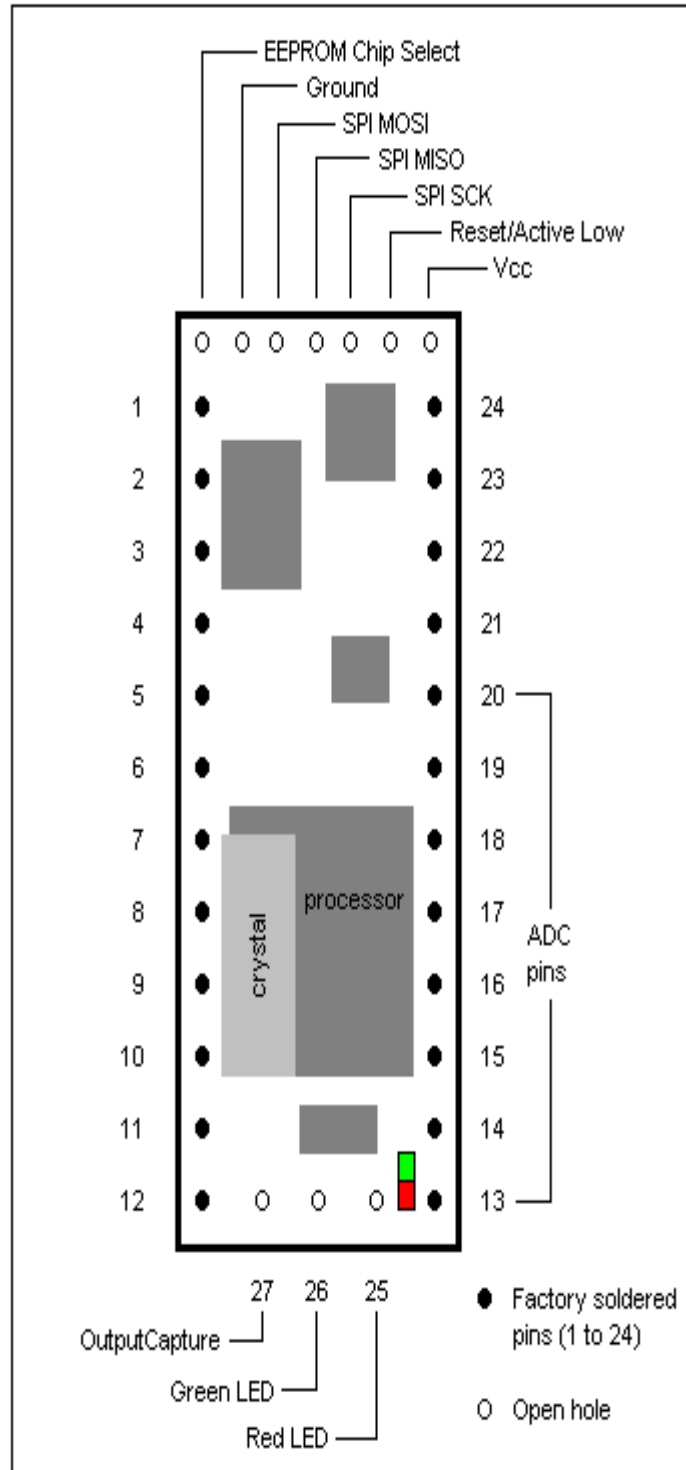
Πίνακας 1	Καταστάσεων Mealy	13
Πίνακας 2	Καταστάσεων (Nondeterministic).....	14
Πίνακας 3	Καταστάσεων Moore.....	14
Πίνακας 4	Κωδικοποίηση δεδομένων.....	19
Πίνακας 5	Κωδικοποίηση καταστάσεων.....	20
Πίνακας 6	Κωδικοποίηση μηχανών Moore και Mealy.....	20
Πίνακας 7	Κατάσταση OFF.....	33
Πίνακας 8	Κατάσταση ON.....	34
Πίνακας 9	Κατάσταση ON (S).....	35
Πίνακας 10	Κατάσταση OFF (S).....	36
Πίνακας 11	Συνολικός Πίνακας.....	37
Πίνακας 12	Κωδικοποιημένος Πίνακας	39

ΠΑΡΑΡΤΗΜΑ Γ

BX-24 technical specifications

I/O Lines	16 total; 8 digital plus 8 lines that can be ADC or digital
EEPROM for program and data storage	On-board 32 KB EEPROM Largest executable user program size is 32 KBytes
RAM	400 bytes
Analog to digital converter	8 channels of 10 bit ADC, can also be used as regular digital (TTL level) I/O
ADC sample rate	6 k samples/s maximum
On-chip LEDs	Has a 2-color surface mount LED (red/green), fully user programmable, not counted as I/O line
Program execution speed	60 microseconds per 16 bit integer add/subtract
Serial I/O speed	2400 baud to 460.8 Kbaud on Com1 300 baud to 19 200 baud on any I/O pin (Com3)
Operating voltage range Min/Max	4.8 VDC to 15.0 VDC
Current requirements	20 mA plus I/O loads, if any
I/O output source current	10 mA @ 5 V (I/O pin driven high)
I/O output sink current	20 mA @ 5 V (I/O pin pulled low)
Combined maximum current load allowed across all I/Os	80 mA sink or source
I/O internal pull-up resistors	120 k Ω maximum
Floating point math	Yes
On-chip multitasking	Yes
On-chip clock/calendar	Yes
Built-in SPI interface	Yes
PC programming interface	Parallel or serial downloads
Package type	24 pin PDIP carrier board
Environmental specifications Absolute maximum ratings	Operating temperature: 0 °C to +70 °C Storage temperature: -65 °C to +150 °C

BX-24 pin numbering



BX-24 Pin Definitions

Each pin on the BX-24 computer has a primary and alternate function as shown in the table below. The primary function describes how the pin can be configured. The alternate function describes how the pin is configured when BasicX built-in options are selected.

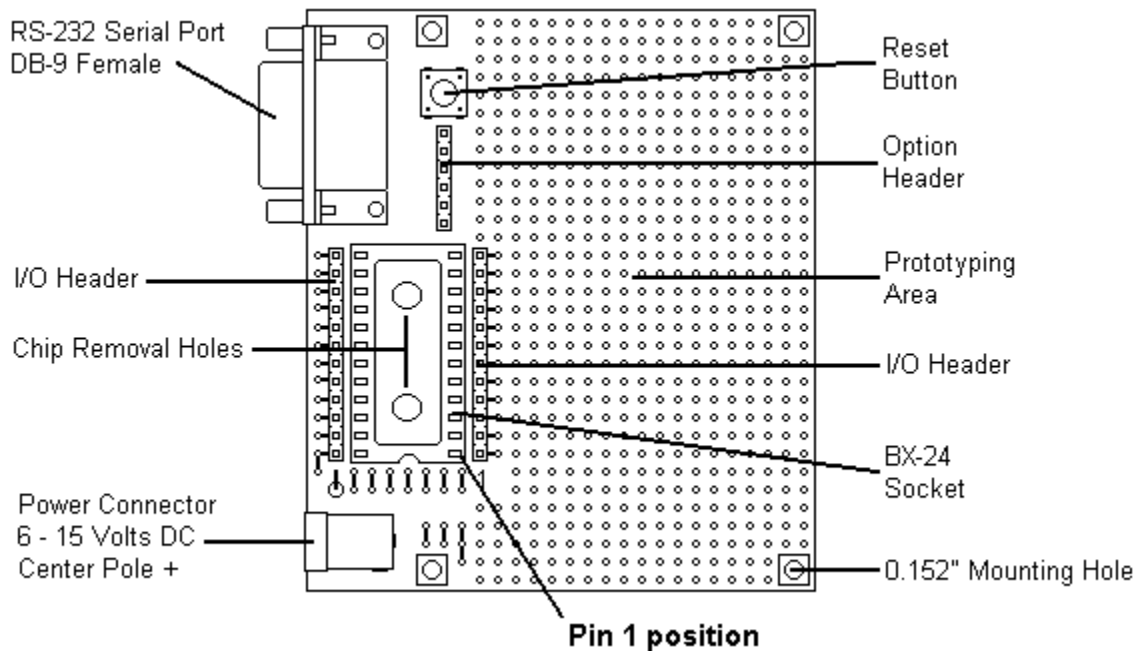
Pin #	Primary Function	Primary Description	Alternate Function	Alternate Description
1	Com1 Transmit	High speed Com port	Serial download transmit	See Com Ports
2	Com1 Receive	High speed Com port	Serial download receive	See Com Ports
3	ATN line	For serial downloading		
4	Ground	Common with pin 23		See below for DC characteristics
5	PortC, Bit 7	General Purpose I/O Port		
6	PortC, Bit 6	General Purpose I/O Port		
7	PortC, Bit 5	General Purpose I/O Port		
8	PortC, Bit 4	General Purpose I/O Port		
9	PortC, Bit 3	General Purpose I/O Port		
10	PortC, Bit 2	General Purpose I/O Port		
11	PortC, Bit 1	General Purpose I/O Port	Interrupt pin	
12	PortC, Bit 0	General Purpose I/O Port	Input capture pin	
13	PortA, Bit 7	ADC channel 7	General Purpose I/O Port	
14	PortA, Bit 6	ADC channel 6	General Purpose I/O Port	
15	PortA, Bit 5	ADC channel 5	General Purpose I/O Port	
16	PortA, Bit 4	ADC channel 4	General Purpose I/O Port	
17	PortA, Bit 3	ADC channel 3	General Purpose I/O Port	
18	PortA, Bit 2	ADC channel 2	General Purpose I/O Port	
19	PortA, Bit 1	ADC channel 1	General Purpose I/O Port	

	1		Port	
20	PortA, Bit 0	ADC channel 0	General Purpose I/O Port	
21	VCC	5 V output from regulator (when powered at Vin)	4.8 V to 5.5 V power input (when Vin not used)	
22	Reset	Low Active Reset		
23	Ground	Common with Pin 4		See below for DC characteristics
24	Vin, 5.5 V to 15 V	Input to 5 V regulator (see note below)		See below for DC characteristics

BX-24 DC characteristics

Parameter	Condition	Min	Typ	Max
Pin 24 power input to voltage regulator	Cannot be used as input if pin 21 is connected	5.5 V	6.0 V	15.0 V
Pin 21 regulated input/output	Cannot be used as input if pin 24 is connected	4.8 V	5.0 V	5.5 V
Ground is common to pins 4 and 23				
Output low voltage	Vcc = 5.0 V Output current = 10 mA			0.5 V
Output high voltage	Vcc = 5.0 V Output current = 10 mA	4.5 V		
Output source current	Vcc = 5.0 V Vcc = 2.7 V			10 mA 5 mA
Output sink current	Vcc = 5.0 V Vcc = 2.7 V			15 mA 10 mA
Maximum total	for all output pins			70 mA
Analog comparator input offset voltage	Vcc = 5.0 V			40 mV
Analog comparator input leakage A	Vcc = 5.0 V Vin = Vcc/2	-50 nA	50 nA	
Analog comparator propagation delay	Vcc = 5.0 V		500 ns	
I/O pin programmable pull-up resistor		35 kΩ		120 kΩ

BX-24 Development Board



Powering the Development board

The BX-24 development board can be powered by a center pole positive 6 VDC to 15 VDC wall transformer with a minimum output rating of 100 mA.

Serial Port Connector

Once you've downloaded a program to the BX-24 chip, the development board's DB-9 serial connector may then be used for other purposes. If you do use the connector for anything other than downloading your program, you should remove Jumper 1 from the option header. By removing Jumper 1 you disconnect the BX-24's ATN line (pin 3) from the serial connector and avoid accidentally putting the BX-24 in download mode.

Listed below are the connections needed to add a DB-9 serial downloading connector to your own BX-24 project.

DB-9 Connector		BX-24 Chip
Pin 2 = TX	>-----<	Pin 1 of the BX-24
Pin 3 = RX	>-----<	Pin 2 of the BX-24
Pin 4 = ATN	>-----<	Pin 3 of the BX-24
Pin 5 = Ground	>-----<	Pins 4 or 23 of the BX-24

Reset Button

Momentarily pushing the reset button causes the BX-24 to halt and restart its stored program from the beginning.

Chip Removal Holes

To aid in the removal of the BX-24 chip, two chip removal holes have been installed underneath the BX-24 socket. (Note: never install or remove your BX-24 chip while power is connected.)

LM78XX Series Voltage Regulators

General Description

The LM78XX series of three terminal regulators is available with several fixed output voltages making them useful in a wide range of applications. One of these is local on card regulation, eliminating the distribution problems associated with single point regulation. The voltages available allow these regulators to be used in logic systems, instrumentation, HiFi, and other solid state electronic equipment. Although designed primarily as fixed voltage regulators these devices can be used with external components to obtain adjustable voltages and currents.

The LM78XX series is available in an aluminum TO-3 package which will allow over 1.0A load current if adequate heat sinking is provided. Current limiting is included to limit the peak output current to a safe value. Safe area protection for the output transistor is provided to limit internal power dissipation. If internal power dissipation becomes too high for the heat sinking provided, the thermal shutdown circuit takes over preventing the IC from overheating.

Considerable effort was expended to make the LM78XX series of regulators easy to use and minimize the number

of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

For output voltage other than 5V, 12V and 15V the LM117 series provides an output voltage range from 1.2V to 57V.

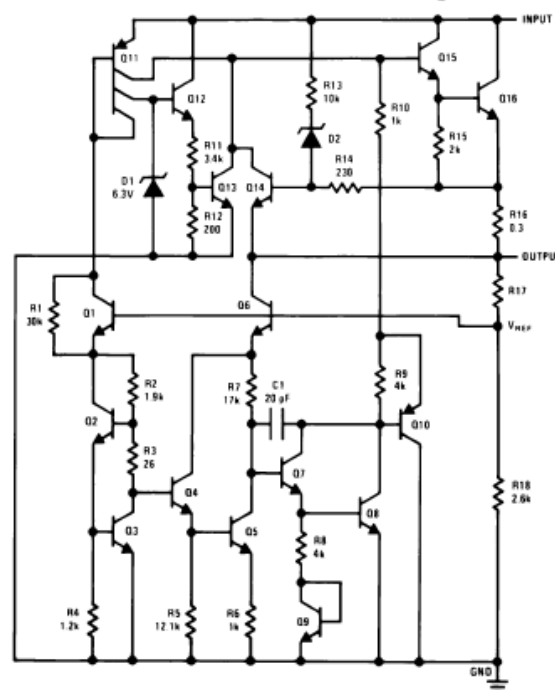
Features

- Output current in excess of 1A
- Internal thermal overload protection
- No external components required
- Output transistor safe area protection
- Internal short circuit current limit
- Available in the aluminum TO-3 package

Voltage Range

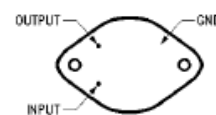
LM7805C	5V
LM7812C	12V
LM7815C	15V

Schematic and Connection Diagrams



TL/H/7746-1

Metal Can Package
TO-3 (K)
Aluminum

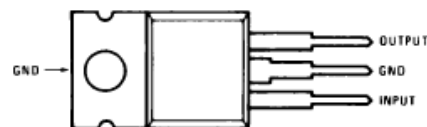


TL/H/7746-2

Bottom View

Order Number LM7805CK,
LM7812CK or LM7815CK
See NS Package Number KC02A

Plastic Package
TO-220 (T)



TL/H/7746-3

Top View

Order Number LM7805CT,
LM7812CT or LM7815CT
See NS Package Number T03B

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Input Voltage ($V_O = 5V, 12V$ and $15V$) 35V
 Internal Power Dissipation (Note 1) Internally Limited
 Operating Temperature Range (T_A) 0°C to $+70^\circ\text{C}$

Maximum Junction Temperature
 (K Package) 150°C
 (T Package) 150°C
 Storage Temperature Range -65°C to $+150^\circ\text{C}$
 Lead Temperature (Soldering, 10 sec.)
 TO-3 Package K 300°C
 TO-220 Package T 230°C

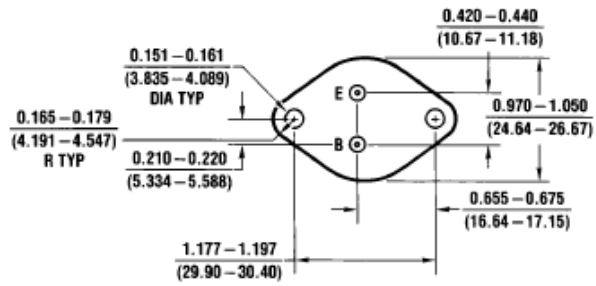
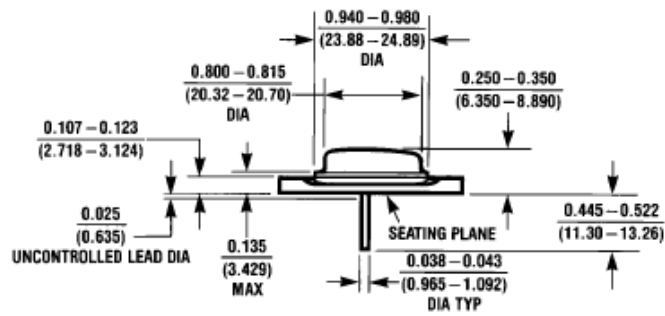
Electrical Characteristics LM78XXC (Note 2) $0^\circ\text{C} \leq T_j \leq 125^\circ\text{C}$ unless otherwise noted.

Output Voltage			5V			12V			15V			Units	
Input Voltage (unless otherwise noted)			10V			19V			23V				
Symbol	Parameter	Conditions	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max		
V_O	Output Voltage	$T_j = 25^\circ\text{C}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.8	5	5.2	11.5	12	12.5	14.4	15	15.6	V	
		$P_D \leq 15\text{ W}, 5\text{ mA} \leq I_O \leq 1\text{ A}$	4.75		5.25	11.4		12.6	14.25		15.75	V	
		$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$	(7.5 $\leq V_{\text{IN}} \leq 20$)			(14.5 $\leq V_{\text{IN}} \leq 27$)			(17.5 $\leq V_{\text{IN}} \leq 30$)			V	
ΔV_O	Line Regulation	$I_O = 500\text{ mA}$	$T_j = 25^\circ\text{C}$	3	50	4	120	4	150			mV	
			ΔV_{IN}	(7 $\leq V_{\text{IN}} \leq 25$)			(14.5 $\leq V_{\text{IN}} \leq 30$)			(17.5 $\leq V_{\text{IN}} \leq 30$)		V	
			$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$		50		120		150		150	mV	
		$I_O \leq 1\text{ A}$	ΔV_{IN}	(8 $\leq V_{\text{IN}} \leq 20$)			(15 $\leq V_{\text{IN}} \leq 27$)			(18.5 $\leq V_{\text{IN}} \leq 30$)		V	
			$T_j = 25^\circ\text{C}$		50		120		150		150	mV	
			ΔV_{IN}	(7.5 $\leq V_{\text{IN}} \leq 20$)			(14.6 $\leq V_{\text{IN}} \leq 27$)			(17.7 $\leq V_{\text{IN}} \leq 30$)		V	
ΔV_O	Load Regulation	$T_j = 25^\circ\text{C}$	$5\text{ mA} \leq I_O \leq 1.5\text{ A}$	10	50	12	120	12	150		mV		
			$250\text{ mA} \leq I_O \leq 750\text{ mA}$		25		60		75		mV		
		$5\text{ mA} \leq I_O \leq 1\text{ A}, 0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$		50		120		150		mV			
I_Q	Quiescent Current	$I_O \leq 1\text{ A}$	$T_j = 25^\circ\text{C}$		8		8		8		8	mA	
			$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$		8.5		8.5		8.5		8.5	mA	
ΔI_Q	Quiescent Current Change	$5\text{ mA} \leq I_O \leq 1\text{ A}$				0.5		0.5		0.5		mA	
		$T_j = 25^\circ\text{C}, I_O \leq 1\text{ A}$				1.0		1.0		1.0		mA	
		$V_{\text{MIN}} \leq V_{\text{IN}} \leq V_{\text{MAX}}$			(7.5 $\leq V_{\text{IN}} \leq 20$)			(14.8 $\leq V_{\text{IN}} \leq 27$)			(17.9 $\leq V_{\text{IN}} \leq 30$)		V
		$I_O \leq 500\text{ mA}, 0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$				1.0		1.0		1.0		mA	
V_N	Output Noise Voltage	$T_A = 25^\circ\text{C}, 10\text{ Hz} \leq f \leq 100\text{ kHz}$				40		75		90		μV	
		ΔV_{IN} ΔV_{OUT}	Ripple Rejection	$f = 120\text{ Hz}$	$I_O \leq 1\text{ A}, T_j = 25^\circ\text{C}$ or	62	80	55	72	54	70		dB
$I_O \leq 500\text{ mA}$	62					55		54			dB		
$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}$	(8 $\leq V_{\text{IN}} \leq 18$)						(15 $\leq V_{\text{IN}} \leq 25$)			(18.5 $\leq V_{\text{IN}} \leq 28.5$)		V	
R_O	Dropout Voltage Output Resistance Short-Circuit Current Peak Output Current Average TC of V_{OUT}	$T_j = 25^\circ\text{C}, I_{\text{OUT}} = 1\text{ A}$			2.0		2.0		2.0			V	
		$f = 1\text{ kHz}$			8		18		19			m Ω	
		$T_j = 25^\circ\text{C}$			2.1		1.5		1.2			A	
		$T_j = 25^\circ\text{C}$			2.4		2.4		2.4			A	
		$0^\circ\text{C} \leq T_j \leq +125^\circ\text{C}, I_O = 5\text{ mA}$			0.6		1.5		1.8			mV/ $^\circ\text{C}$	
V_{IN}	Input Voltage Required to Maintain Line Regulation	$T_j = 25^\circ\text{C}, I_O \leq 1\text{ A}$			7.5		14.6		17.7		V		

Note 1: Thermal resistance of the TO-3 package (K, KC) is typically $4^\circ\text{C}/\text{W}$ junction to case and $35^\circ\text{C}/\text{W}$ case to ambient. Thermal resistance of the TO-220 package (T) is typically $4^\circ\text{C}/\text{W}$ junction to case and $50^\circ\text{C}/\text{W}$ case to ambient.

Note 2: All characteristics are measured with capacitor across the input of $0.22\text{ }\mu\text{F}$, and a capacitor across the output of $0.1\text{ }\mu\text{F}$. All characteristics except noise voltage and ripple rejection ratio are measured using pulse techniques ($t_w \leq 10\text{ ms}$, duty cycle $\leq 5\%$). Output voltage changes due to changes in internal temperature must be taken into account separately.

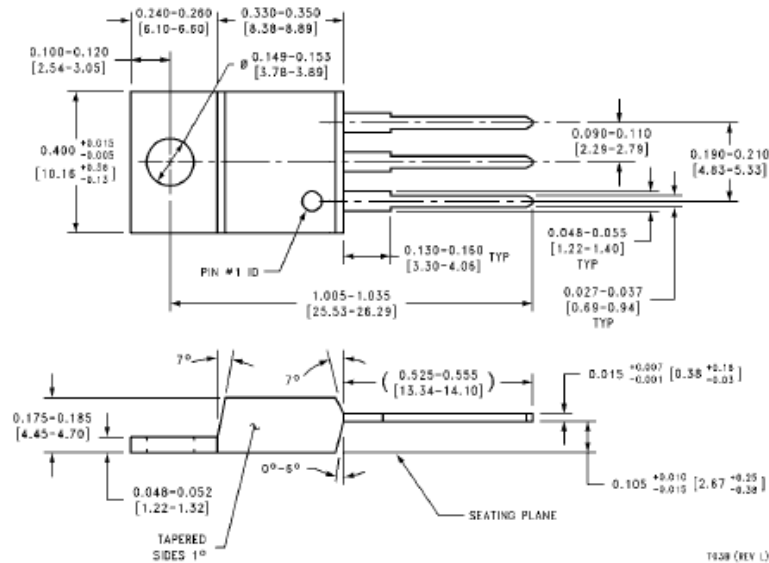
Physical Dimensions inches (millimeters)



KC02A (REV C)

Aluminum Metal Can Package (KC)
 Order Number LM7805CK, LM78 12CK or LM7815CK
 NS Package Number KC02A

Physical Dimensions inches (millimeters) (Continued)



TO-220 Package (T)
Order Number LM7805CT, LM7812CT or LM7815CT
NS Package Number T03B

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.



National Semiconductor Corporation
 1111 West Bardin Road
 Arlington, TX 76017
 Tel: 1(800) 272-9959
 Fax: 1(800) 737-7018

National Semiconductor Europe
 Fax: (+49) 0-180-530 85 86
 Email: cnjwge@lvm2.nsc.com
 Deutsch Tel: (+49) 0-180-530 85 85
 English Tel: (+49) 0-180-532 78 32
 Français Tel: (+49) 0-180-532 93 58
 Italiano Tel: (+49) 0-180-534 18 80

National Semiconductor Hong Kong Ltd.
 13th Floor, Straight Block,
 Ocean Centre, 5 Canton Rd.,
 Tsimshatsui, Kowloon
 Hong Kong
 Tel: (852) 2737-1600
 Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
 Tel: 81-043-299-2309
 Fax: 81-043-299-2408

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.