



ΑΝΩΤΑΤΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ ΤΕΧΝΟΛΟΓΙΚΟΥ ΤΟΜΕΑ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ Τ.Ε

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ενσωματωμένα Συστήματα στη Βιομηχανία

Σπουδαστής:
Καφετζηδάκης Αντώνιος

Επιβλέπουσα:
Βελώνη Αναστασία

Αθήνα 2017

ΠΕΡΙΛΗΨΗ

Αντικείμενο της παρούσας πτυχιακής εργασίας αποτελεί η μελέτη και η ανάλυση των ενσωματωμένων συστημάτων (embedded systems), καθώς και η ανασκόπηση της εφαρμογής αυτών στη βιομηχανία. Στόχο της εν λόγω εργασίας αποτελεί αφενός η πλήρης κατανόηση του ορισμού του ενσωματωμένου συστήματος και των επιμέρους τμημάτων που το απαρτίζουν, αφετέρου η ανάδειξη της συμβολής τους στα σύγχρονα βιομηχανικά συστήματα.

Για την επίτευξη του παραπάνω σκοπού, η διάρθρωση της εργασίας αρχικά περιλαμβάνει το εισαγωγικό κεφάλαιο, όπου γίνεται αναφορά στα ψηφιακά συστήματα, ως αναπόσπαστο κομμάτι των ενσωματωμένων συστημάτων, δίδεται ο ορισμός του ενσωματωμένου συστήματος, καθώς και κάποια παραδείγματα για την αποσαφήνιση του ορισμού αυτού. Στη συνέχεια, ο κύριος κορμός πραγματεύεται ζητήματα αρχιτεκτονικής, σχεδιασμού, υλοποίησης, μοντελοποίησης, ψηφιακών συστημάτων, μικροελεγκτών-ελεγκτών, προγραμματισμού, χρήσης υλικού (hardware) και λογισμικού (software), προβλημάτων, κ.ά. Παράλληλα, σε κάθε κεφάλαιο παρατίθενται παραδείγματα, με σκοπό την αποσαφήνιση του θεωρητικού τμήματος και της σύνδεσής του με την χρήση των εν λόγω συστημάτων στη βιομηχανία και κατ' επέκταση στην καθημερινότητα.

Τέλος, παρουσιάζονται και αναλύονται ορισμένα σύγχρονα ενσωματωμένα συστήματα, τα οποία βρίσκουν εφαρμογή σε ευρύ φάσμα τόσο της βιομηχανίας, όσο και του τομέα έρευνας και ανάπτυξης (R&D) σε πανεπιστημιακά ιδρύματα και ερευνητικά κέντρα σε παγκόσμια κλίμακα. Με αυτόν τον τρόπο αποτυπώνεται εμπειριστικά η αξία και η συμβολή των ενσωματωμένων συστημάτων στην

ανάπτυξη καινοτόμων τεχνολογιών, με εφαρμογή σε όλους του τομείς της ανθρώπινης δραστηριότητας.

Λέξεις κλειδιά: Ενσωματωμένα συστήματα, ψηφιακά συστήματα, επεξεργαστές, γλώσσες προγραμματισμού, μικροελεγκτές, βιομηχανία, ανανεώσιμες πηγές ενέργειας

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ	3
ΚΕΦΑΛΑΙΟ 1: ΟΡΙΣΜΟΣ ΤΟΥ ΕΝΣΩΜΑΤΩΜΕΝΟΥ ΣΥΣΤΗΜΑΤΟΣ [1], [2], [3].....	8
1.1 Εισαγωγή	8
1.2 Τα ψηφιακά συστήματα	8
1.2.1 Σύντομη Ιστορική ανασκόπηση των ψηφιακών συστημάτων [1].....	8
1.2.1.1 Πρώτη Γενιά	11
1.2.1.2 Δεύτερη Γενιά	12
1.2.1.3 Τρίτη Γενιά	12
1.2.1.4 Τέταρτη Γενιά	13
1.2.1.5 Πέμπτη Γενιά	14
1.3 Τα ενσωματωμένα συστήματα.....	15
1.3.1 Παραδείγματα ενσωματωμένων συστημάτων	17
1.3.2 Ορισμός του ενσωματωμένου συστήματος.....	19
1.4 Εισαγωγή στην αγορά του ενσωματωμένου συστήματος.....	24
1.4.1 Κοστολόγηση [1]	24
1.4.2 Εισαγωγή του προϊόντος στην αγορά	26
1.4.3 Εμπορικές και κατασκευαστικές προδιαγραφές των Ε.Σ.....	29
ΚΕΦΑΛΑΙΟ 2: ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΕΝΣΩΜΑΤΩΜΕΝΩΝ ΣΥΣΤΗΜΑΤΩΝ [6].....	34
2.1 Εισαγωγή	34
2.2 Αναγκαιότητα της μοντελοποίησης	35
2.2.1 Σημασία της μοντελοποίησης	35
2.2.2 Οικονομικά στοιχεία μοντελοποίησης.....	38
2.2.3 Εξάρτηση του NRE από τη μοντελοποίηση	43
2.2.4 Αρχές μοντελοποίησης.....	44
2.3 Περιβάλλον μοντελοποίησης.....	48
ΚΕΦΑΛΑΙΟ 3: ΟΡΓΑΝΩΣΗ ΚΑΙ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΩΝ Ε.Σ.	52
3.1 Εισαγωγή	52
3.2 Η μνήμη	52
3.2.1 Μνήμη τυχαίας προσπέλασης.....	56
3.2.2 Μνήμη Ανάγνωσης μόνο	63

3.2.3 Erasable Programmable Memory.....	65
3.2.4 Electrically Erasable Programmable Memory	66
3.2.5 Η μνήμη Flash.....	67
3.2.6 Αποθήκευση των Byte	68
3.2.7 Ευθυγραμμισμένες προσβάσεις στα Δεδομένα.....	73
3.2.8 Κρυφή Μνήμη [1], [5]	78
3.3 Ο επεξεργαστής ενός Ε.Σ.....	80
3.4 Αρχιτεκτονική αποθηκευμένου προγράμματος	81
3.4.1 Κώδικας μηχανής.....	86
3.4.2 Απαιτήσεις μείωσης της κατανάλωσης ενέργειας του συστήματος	88
ΚΕΦΑΛΑΙΟ 4 : ΠΑΡΑΔΕΙΓΜΑΤΑ Ε.Σ. ΣΤΗ ΒΙΟΜΗΧΑΝΙΑ.....	89
4.1 Παράδειγμα πρώτο: Το Ε.Σ. μιας επαγωγικής κουζίνας [15]	90
4.2 Παράδειγμα δεύτερο: Το Ε.Σ. για ένα ψηφιακό καταγραφέα [15]	96
4.3 Παράδειγμα Τρίτο: Το Ε.Σ. για ένα αυτοκίνητο [15]	100
4.3.1 Το σύστημα του αερόσακου [15].....	105
4.3.2 Event Data Recorders (EDRs)	109
4.3.3 Σύστημα Αντιμπλοκαρίσματος Τροχών ABS (Anti-lock Brake System)	113
ΚΕΦΑΛΑΙΟ 5: ΣΥΜΠΕΡΑΣΜΑΤΑ	117
ΒΙΒΛΙΟΓΡΑΦΙΑ	120

ΚΕΦΑΛΑΙΟ 1: ΟΡΙΣΜΟΣ ΤΟΥ ΕΝΣΩΜΑΤΩΜΕΝΟΥ ΣΥΣΤΗΜΑΤΟΣ [1], [2], [3]

1.1 Εισαγωγή

Ένα σπουδαίο απότοκο της ψηφιακής εποχής και των δυνατοτήτων που δίδονται από τη χρήση των ηλεκτρονικών υπολογιστών είναι τα ενσωματωμένα συστήματα (embeddedsystems). Πρόκειται για συστήματα τα οποία απαιτούν κάποιον έλεγχο, λόγου χάρη ένας οικιακός αυτοματισμός, τον οποίο καλείται να εκτελέσει ένας υπολογιστής που έχει ενσωματωθεί σε αυτά. Πρόκειται δηλαδή για ένα σύστημα του οποίου η κύρια λειτουργία είναι μη υπολογιστική και είναι ελεγχόμενο από έναν ενσωματωμένο σε αυτό υπολογιστή.

1.2 Τα ψηφιακά συστήματα

1.2.1 Σύντομη Ιστορική ανασκόπηση των ψηφιακών συστημάτων [1]

Είτε πρόκειται για πληροφοριακά συστήματα, είτε για συστήματα όπου η κύρια λειτουργία τους είναι μη υπολογιστική (όπως τα ενσωματωμένα συστήματα), οι ψηφιακοί υπολογιστές διαδραματίζουν καθοριστικό ρόλο στην ανάπτυξή τους. Ωστόσο, καταλυτικό παράγοντα για την εξέλιξη των συστημάτων αυτών αποτέλεσε η εξέλιξη των ίδιων των ψηφιακών συστημάτων, γι' αυτό και μια ανασκόπηση στην εξέλιξη των ψηφιακών συστημάτων στο σημείο αυτό κρίζει σκόπιμη για την πληρέστερη κατανόηση και ανάλυση των ενσωματωμένων συστημάτων.

Οι σύγχρονοι ηλεκτρονικοί υπολογιστές (όπως αυτοί που περιλαμβάνονται στα

ενσωματωμένα συστήματα), γνώρισαν μεγάλη εξέλιξη το διάστημα από το 1940 μέχρι σήμερα. Πρωτότερα, χρησιμοποιούνταν μηχανές οι οποίες δεν ήταν ψηφιακές αλλά χρησιμοποιούσαν μηχανικά τμήματα που αποτελούνταν από πολύπλοκες κατασκευές που περιλάμβαναν μοχλούς, γρανάζια, ταινίες, διακόπτες, κ.ά. και χρησιμοποιούνταν για τον υπολογισμό των βασικών πράξεων, δηλαδή της πρόσθεσης, της αφαίρεσης, του πολλαπλασιασμού και της διαίρεσης. Επιπλέον, ορισμένες από αυτές τις μηχανές προγραμματίζονταν με τη χρήση διάτρητων καρτών -καρτών που με οπές σε συγκεκριμένα σημεία- να προσδιορίζουν μια συγκεκριμένη αλληλουχία πράξεων, γεγονός που προσομοιώνει κατά κάποιο τρόπο τον προγραμματισμό σε γλώσσα μηχανής (assemblycode) ο οποίος χρησιμοποιείται σήμερα σε εφαρμογές όπου απαιτείται προγραμματισμός χαμηλού επιπέδου (στο επίπεδο της μηχανής). Ένα παράδειγμα κατά κόρον χρήσης τέτοιου είδους προγραμματισμού αποτελούν οι υφαντικές μηχανές.

Ωστόσο, η διαρκώς αυξανόμενη ανάγκη για εξελιγμένη τεχνολογία, οδήγησε στην ανάπτυξη κατασκευών με βάση το ψηφιακό σύστημα (σύστημα δυο καταστάσεων, την ενεργή (on)και την ανενεργή(off)), οι οποίες άρχισαν να εμφανίζονται στα τέλη της δεκαετίας του 1930. Για μια σειρά λόγους (Β' Παγκόσμιος Πόλεμος, κάλυψη στρατιωτικών αναγκών, αξιοπιστία των επικοινωνιών κατά την περίοδο του πολέμου), επιστήμονες, φυσικοί, μαθηματικοί και μηχανικοί από κάθε αντιμαχόμενο μπλοκ, κατασκεύασαν υπολογιστικές μηχανές για την κρυπτογράφηση και αποκρυπτογράφηση μηνυμάτων, τον υπολογισμό της τροχιάς των βλημάτων και τη βελτίωση των επικοινωνιών. Σύμφωνα με τις επίσημες πηγές, η πρώτη υπολογιστική μηχανή που επέτυχε να δώσει ένα προβάδισμα στους συμμάχους με την αποκρυπτογράφηση των γερμανικών επικοινωνιών, είναι το Colossus που κατασκευάστηκε στην Αγγλία το 1943 από μια ομάδα, με επικεφαλής τους Alan Turing και Tommy Flowers. Οι γερμανικές επικοινωνίες χρησιμοποιούσαν την κρυπτογράφηση με την περιβόητη μηχανήEnigma (η οποία

αποτελείτο από γρανάζια και ανήκει στην κατηγορία των μη επαναπρογραμματιζόμενων μηχανών, αντίθετα από τους ηλεκτρονικούς υπολογιστές).

Λίγο αργότερα, κατά το διάστημα του ψυχρού πολέμου, στην άλλη πλευρά του Ατλαντικού Ωκεανού, κατασκευάστηκε ο υπολογιστής ENIAC (Electronic Numerical Integrator And Computer) , για λόγους χρήσης από τον Στρατό για τον υπολογισμό βαλλιστικών τροχιών. Για ιστορικούς λόγους αξίζει να αναφερθεί ότι, καθώς ο Colossus αποτελούσε κρατικό μυστικό και η ύπαρξη του ήταν απόρρητη έως και το 2000, ο ENIAC θεωρείτο ο πρώτος προγραμματιζόμενος υπολογιστής έως τότε, με πολλά ακαδημαϊκά συγγράμματα να αναφέρουν μόνο αυτόν. Ανάμεσα στους δυο αυτούς πρώτους υπολογιστές, τον ENIAC και τον Colossus, διακρίνεται μια ουσιαστική διαφορά: ο μεν πρώτος χρησιμοποιούσε το δεκαδικό σύστημα σε μια μορφή που είναι γνωστή ως BCD (binary coded decimal), ο δε άλλος χρησιμοποιούσε το δυαδικό σύστημα (binary, 01).

Πέραν των προαναφερθέντων, όπως ήταν επόμενο, κατασκευάστηκαν και άλλοι υπολογιστές οι οποίοι όμως είτε δεν ήταν προγραμματιζόμενοι είτε χρησιμοποιούσαν ηλεκτρονόμους (relay). Σταθμό για τη μεγάλη επανάσταση στη βιομηχανία των ψηφιακών συστημάτων, αποτέλεσε η εφεύρεση του τρανζίστορ (1947) [4]. Το ημιαγωγικό αυτό στοιχείο τριών ακροδεκτών (ο ένας εκ των οποίων, η βάση, ρυθμίζει το ρεύμα (τη ροή ηλεκτρονίων) που διέρχεται διαμέσου των δυο άλλων επαφών, του συλλέκτη και του εκπομπού) έχει τη δυνατότητα να επιτρέπει ή όχι τη ροή των ηλεκτρονίων, δημιουργώντας τις καταστάσεις on και off που αναφέρθηκαν παραπάνω αντίστοιχα. Ο πρώτος υπολογιστής στον οποίο χρησιμοποιήθηκαν τα τρανζίστορ ήταν ο Transistor Computer από το Πανεπιστήμιο του Manchester. Τα επόμενα χρόνια πλήθος υπολογιστών αναπτύχθηκαν από τα πανεπιστήμια σε παγκόσμια κλίμακα, όπως το MIT, του Cambridge, του Manchester κ.ά., καθώς και από πασίγνωστες και πρωτοπόρες

εταιρίες, όπως οι Intel, IBM, DEC, ARM, CRAY, κτλ., με πληθώρα δυνατοτήτων και χαρακτηριστικών. Λόγω των διαφορετικών τεχνολογικών και φιλοσοφιών που χρησιμοποίησαν οι μηχανικοί και επιστήμονες για τη δημιουργία των υπολογιστών, είθισται η κατηγοριοποίηση της εξέλιξής τους σε πέντε γενιές[1], [5].

1.2.1.1 Πρώτη Γενιά

Στην πρώτη γενιά κατατάσσονται οι υπολογιστές που χρησιμοποιούν λυχνίες κενού.

Τα χαρακτηριστικά των υπολογιστών που ανήκουν σε αυτήν την γενιά είναι τα ακόλουθα:

- Μικρός χρόνος ανάμεσα σε σφάλματα (MTBF, mean time between failures)
- Χρήση του δεκαδικού συστήματος(BCD)
- Χρήση καλωδίων, διακοπών και περιστροφικών γραναζιών για τον προγραμματισμό
- Προγραμματισμός σε γλώσσα χαμηλού επιπέδου (στο επίπεδο της μηχανής-assembly).
- Μεγάλο μέγεθος (συνήθως όσο ένα μεγάλο δωμάτιο)
- Χαμηλή ταχύτητα επεξεργασίας

Οι γνωστότεροι εκπρόσωποι αυτής της γενιάς υπολογιστών είναι οι προαναφερθέντες Colossus και ENIAC, με τον δεύτερο να ζυγίζει 30 τόνους και να καταλαμβάνει 1300 τετραγωνικά μέτρα, ενώ η ταχύτητά του περιορίζονταν στις 500 προσθέσεις το δευτερόλεπτο.

1.2.1.2 Δεύτερη Γενιά

Η δεύτερη γενιά χρησιμοποιεί τρανζίστορ μεγάλο μεγέθους, καθιστώντας τους υπολογιστές αυτούς αρκετά ογκώδη μηχανήματα. Τα χαρακτηριστικά της εν λόγω γενιάς συνοψίζονται στα εξής:

- Βελτιωμένη αξιοπιστία χάρις στα τρανζίστορ, στα οποία χρησιμοποιείται το δυαδικό σύστημα (για τις δυο καταστάσεις που διαθέτουν τα τρανζίστορ, οπια λογικό 1 και off για λογικό 0)
- Χρησιμοποιούν διάτρητες κάρτες
- Χρησιμοποιούν πρώτες γλώσσες προγραμματισμού υψηλού επιπέδου (στο επίπεδο του ανθρώπου).

Το πιο γνωστό μηχάνημα αυτής της γενιάς είναι ο PDP-1 της εταιρίας DEC, ο οποίος διέθετε μνήμη 4KB και συχνότητα ρολογιού στα 0.2 MHz, ενώ η αρχική του τιμή ήταν \$100KUSD. Ένα από τα σημαντικά πλεονεκτήματά του ήταν η επεκτασιμότητα του συστήματος, καθώς ο χρήστης μπορούσε να προμηθευτεί επιπρόσθετα, πληκτρολόγιο, εκτυπωτή, αποθηκευτικό χώρο, οθόνη και κάμερα.

1.2.1.3 Τρίτη Γενιά

Χαρακτηριστικά της τρίτης γενιάς:

- Αυξημένη αξιοπιστία (βελτιωμένη τεχνολογία)
- Χρήση λειτουργικών συστημάτων με δυνατότητα ταυτόχρονης εκτέλεσης πολλαπλών προγραμμάτων (ψευδό-παράλληλα), μέσω χρονοδρομολογητή
- Βελτιωμένη χρήση των υπολογιστικών πόρων (πολλαπλοί χρήστες μοιράζονται τους πόρους ταυτόχρονα)

- Χρήση ολοκληρωμένων κυκλωμάτων ημιαγωγού πυριτίου. Πρόκειται για κυκλώματα με πολλά τρανζίστορ που κατασκευάζονται στην ίδια ψηφίδα πυριτίου με φωτολιθογραφικές μεθόδους.
- Χρήση της έννοιας της ιδεατής μνήμης, η οποία επιτρέπει την αποδοτικότερη εκμετάλλευση της ελάχιστης μνήμης τυχαίας προσπέλασης(RAM), με τη χρήση τεχνικών σελιδοποίησης και επέκτασης σε δευτερεύουσα μνήμη (σκληρό δίσκο).

Το πιο αντιπροσωπευτικό παράδειγμα υπολογιστή αυτής της κατηγορίας είναι το μοντέλο IBM System/360 που διέθετε 512KB RAM, με συχνότητα ρολογιού στα 4MHz (20 φορές ταχύτερο ρολόι σε σχέση με της προηγούμενης γενιάς υπολογιστές). Ο επεξεργαστής που χρησιμοποιούσε χρησιμοποιούσε καταχωρητές (registers) και είχε διασωλήνωση (pipelining), ενώ χρησιμοποιούσε μικροκώδικα μέσα στους επεξεργαστές για την εξασφάλιση της συμβατότητας με άλλα μηχανήματα. Αυτός ο υπολογιστής, που είχε το μέγεθος ενός μεγάλου διπλού ψυγείου, χρησιμοποιήθηκε σε μια πιο «ενσωματωμένη» έκδοση (σε μέγεθος μιας βαλίτσας) για να εγκατασταθεί στο διαστημόπλοιο Apollo 11 που προσελήνωσε η NASA το 1969. Ωστόσο, χάριν αύξησης της αξιοπιστίας του συστήματος, η IBM με 4000 μηχανικούς, επανασχεδίασε το ενσωματωμένο αυτό σύστημα, με μέσο χρόνο λειτουργίας μεταξύ βλαβών στις 25000 ώρες και πυκνότητα ολοκλήρωσης 40000 στοιχείων ανά κυβικό πόδι (για τρισδιάστατη κατασκευή). Το Apollo Guidance Computer ήταν το πρώτο ενσωματωμένο σύστημα που κατασκευάστηκε.

1.2.1.4 Τέταρτη Γενιά

Στη γενιά αυτή υπάγονται όλοι οι υπολογιστές από τα μέσα της δεκαετίας του 1990 και έπειτα. Τα χαρακτηριστικά τους:

- Μεγάλη αξιοπιστία
- Πολυάριθμες δυνατότητες

- Μεγάλη πυκνότητα ολοκλήρωσης
- Λειτουργικά συστήματα με εύχρηστο περιβάλλον εργασίας για τον άνθρωπο (χρήστη)
- Πληθώρα τεχνικών για την αποδοτικότερη εκμετάλλευση των πόρων του συστήματος, όπως εικονική μνήμη, ψευδό-παραλληλισμός εργασιών, πολλαπλοί ταυτόχρονοι χρήστες, πρωτόκολλα άμεσης πρόσβασης σε υλικό κ.ά.

Τόσο το λογισμικό (software) όσο και το υλικό (hardware) έχουν βελτιωθεί, με την ενσωμάτωση των βέλτιστων τεχνικών των τελευταίων 50 ετών, χρησιμοποιώντας πολύ μεγάλες πυκνότητες ολοκλήρωσης και σύγχρονες μεθόδους επικοινωνίας και διασύνδεσης.

1.2.1.5 Πέμπτη Γενιά

Η Πέμπτη γενιά υπολογιστών δεν έχει εμφανιστεί με αμιγή χαρακτηριστικά. Ουσιαστικά, πρόκειται για μια γενιά για την οποία διατίθενται μόνο προβλέψεις και σχέδια νέων τεχνολογιών και δοκιμών. Στόχο της γενιάς αυτής αποτελούν ακόμη πιο φιλικά στην χρήση λειτουργικά συστήματα όσον αφορά το λογισμικό, ενώ για το υλικό αυξάνουν οι απαιτήσεις προδιαγραφών εμφάνισης. Λόγος γίνεται για υπολογιστές που θα είναι ενσωματωμένοι σε διάφορα συστήματα γενικής και ειδικής χρήσης (ubiquitous computing, πανταχόθεν υπολογιστική δυνατότητα) χωρίς ευκρινή προσδιορισμό. Συνεπώς, ορισμένα χαρακτηριστικά που μπορούν να περιγράψουν αυτήν την κατηγορία αναφέρονται σε υπολογιστές:

- «αθέατους» και διασυνδεδεμένους
- Επικοινωνίας με την ομιλία και όχι με πληκτρολόγιο ή ποντίκι

- Με πλήρη κατάργηση καλωδίων, χρήσης αποκλειστικά ασύρματων συνδέσεων

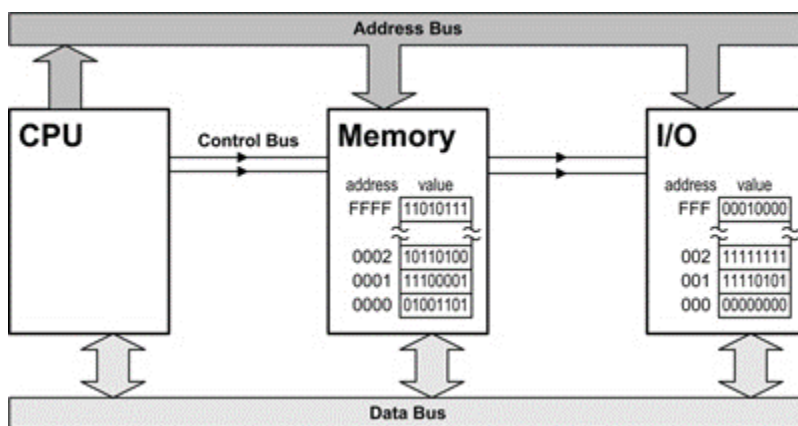
Ένα παράδειγμα αυτής της νέας γενιάς υπολογιστών, που αποτελούν την επιτομή των ενσωματωμένων συστημάτων του παρόντος και του μέλλοντος, αποτελούν οι «έξυπνες» τηλεοράσεις. Σε αυτές καταργείται ολοσχερώς η σύνδεση με έναν ενδιάμεσο υπολογιστή και ο χρήστης διαθέτει αποκλειστικά μια οθόνη αντί ολόκληρου υπολογιστή (στην ουσία πρόκειται για απουσία εξωτερικής κεντρικής μονάδας, αφού είναι ενσωματωμένη μέσα στην οθόνη.) Τα αρχεία, τα προγράμματα και οι εφαρμογές βρίσκονται στο διαδίκτυο (τακτοποιημένες στα λεγόμενα «σύννεφα» (clouds)) και είναι προσβάσιμα από οποιαδήποτε διασυνδεδεμένη συσκευή. Οι συσκευές του χρήστη είναι τερματικές συσκευές για απομακρυσμένη σύνδεση σε κεντρικούς διακομιστές του διαδικτύου.

1.3 Τα ενσωματωμένα συστήματα

Σύμφωνα με όσα αναφέρθηκαν στην προηγούμενη ενότητα, η ειδοποιός διαφορά ανάμεσα στα κοινά υπολογιστικά συστήματα και στα ενσωματωμένα συστήματα είναι ότι οι περισσότεροι χρήστες μπορούν εύκολα να διακρίνουν τα πρώτα από τα φυσικά τους γνωρίσματα, όπως εκείνα ενός επιτραπέζιου υπολογιστή, ενός διακομιστή ή ενός φορητού υπολογιστή. Ωστόσο, στη δεύτερη περίπτωση, τα ενσωματωμένα συστήματα είναι αθέατα (γι' αυτό άλλωστε και καλούνται ενσωματωμένα), γι' αυτό και δεν υπάρχει η δυνατότητα αναγνώρισής τους από το χρήστη. Σήμερα, κατασκευάζονται ετησίως δισεκατομμύρια μονάδες ενσωματωμένων συστημάτων, πλέον σε με πολύ μεγαλύτερο ρυθμό παραγωγής σε σχέση με τις τυπικές υπολογιστικές μονάδες, οι οποίες κυμαίνονται σε κάποια εκατομμύρια. Τη χαρακτηριστικότερη απόδειξη αποτελεί ένα σύγχρονο σπίτι στο

οποίο υπολογίζεται ότι υπάρχουν ένας με δυο ηλεκτρονικοί υπολογιστές και πάνω από 50 ενσωματωμένα συστήματα (Ε.Σ.).

Με τον όρο Ε.Σ. περιγράφεται οποιαδήποτε συσκευή η οποία περιλαμβάνει έναν προγραμματιζόμενο επεξεργαστή, ο οποίος δεν είναι ένας επεξεργαστής γενικού σκοπού, αλλά προορίζεται να επιτελεί συγκεκριμένη λειτουργία. Τα ενσωματωμένα συστήματα έχουν ίδια αρχιτεκτονική με τα τυπικά υπολογιστικά συστήματα. Όπως κάθε προγραμματιζόμενο ψηφιακό σύστημα, έτσι και αυτά διαθέτουν έναν ή περισσότερους επεξεργαστές, μνήμη τυχαίας προσπέλασης (RAM) και μνήμη ανάγνωσης μόνο (ROM) καθώς και διεπαφές εισόδου/εξόδου. Η διαφοροποίηση των ενσωματωμένων συστημάτων έγκειται στη χρήση αποκλειστικά μικροεπεξεργαστών, με περισσότερες δυνατότητες διασύνδεσης εισόδου/εξόδου και λιγότερους πόρους (λόγου χάρη μικρότερη μνήμη). Με άλλα λόγια, ένα ενσωματωμένο σύστημα μπορεί να θεωρηθεί ως μια μικρογραφία ενός τυπικού υπολογιστικού συστήματος. Σε επόμενες ενότητες συζητούνται οι διαφορετικές απαιτήσεις των Ε.Σ., που έχουν οδηγήσει στη διαφοροποίησή τους από τα τυπικά υπολογιστικά συστήματα.



Εικόνα 1-Ένα Ε.Σ. το οποίο, όπως και τα τυπικά υπολογιστικά συστήματα, διαθέτει έναν επεξεργαστή, μία μνήμη και τουλάχιστον έναν τρόπο επικοινωνίας με το περιβάλλον (είσοδο/έξοδο). [1], [5]

1.3.1 Παραδείγματα ενσωματωμένων συστημάτων

Τα ενσωματωμένα συστήματα αρχικά, είχαν εμφανιστεί ως συσκευές ή προϊόντα χαμηλών επιδόσεων, λόγω χάρη ρολόγια και ηλεκτρονικοί υπολογιστές τσέπης. Ωστόσο σήμερα, στις τεχνολογικά αναπτυγμένες κοινωνίες, περιλαμβάνονται σε αρκετά πολύπλοκες συσκευές με αυξημένες απαιτήσεις επιδόσεων, όπως είναι τα φορητά τερματικά πολυμέσων, οι ψηφιακοί προσωπικοί βοηθοί (PDA), κ.ά. Έτσι, τα ενσωματωμένα συστήματα έχουν γίνει πλέον μέρος της καθημερινής ζωής των περισσότερων ανθρώπων, ακόμα κι αν είναι όπως προαναφέρθηκε «αθέατα» σε αυτούς.

Ενσωματωμένα συστήματα εξειδικευμένων εφαρμογών (περιορισμένης χρήσης, ενσωματώνονται στις ψηφιακές συσκευές) απαντώνται στις περισσότερες εφαρμογές της καθημερινότητας. Στο σημείο αυτό μπορεί να αναφερθεί ένας ακόμη γενικός ορισμός του ενσωματωμένου συστήματος, ο οποίος περιγράφει κάθε ενσωματωμένο σύστημα ως ένα ψηφιακό σύστημα που εκτελεί κάποια λειτουργία και δεν είναι προσωπικός υπολογιστής, φορητός υπολογιστής ή κεντρικός διακομιστής (mainframe). Στη συνέχεια παρατίθενται ορισμένα παραδείγματα, για την καλύτερη κατανόηση του τι είναι ένα ενσωματωμένο σύστημα:

- Τηλεοράσεις και βίντεο. Οι τηλεοράσεις έχουν ενσωματωμένους επεξεργαστές για να ελέγχουν την εικόνα, να ρυθμίζουν τα κανάλια, να εκτυπώνουν μηνύματα πάνω στην εικόνα, να απενεργοποιούν και να ενεργοποιούν τα κυκλώματα της τηλεόρασης, κτλ.
- Συσκευές τηλε-ελέγχου. Οι συσκευές τηλε-ελέγχου (telecontrol) έχουν

ενσωματωμένους επεξεργαστές για να μετατρέπουν τις εντολές του χρήστη σε σήματα υπέρυθρων για να ελέγχουν άλλες συσκευές.

- **Κινητά/Ασύρματα τηλέφωνα.** Τα κινητά τηλέφωνα έχουν ενσωματωμένους επεξεργαστές για να διεκπεραιώνουν τις διαδικασίες αποκωδικοποίησης/κωδικοποίησης της φωνής, για να υλοποιούν το πρωτόκολλο επικοινωνίας και να εκτελούν τις πολύπλοκες εφαρμογές του τηλεφώνου.
- **Οχήματα.** Όλα τα σύγχρονα αυτοκίνητα έχουν ένα πλήθος από ενσωματωμένους επεξεργαστές, είτε για την προστασία των επιβατών (σύστημα πέδησης ABS), είτε για βοηθητικές ενδείξεις, είτε για τη βελτίωση των συνθηκών οδήγησης.
- **Συσκευές νοικοκυριού.** Οι σύγχρονες συσκευές νοικοκυριού (φούρνοι, πλυντήρια, ψυγεία κ.ά.) διαθέτουν ενσωματωμένους επεξεργαστές, για να προστατεύουν τα τρόφιμα ή τα ρούχα από λανθασμένες ρυθμίσεις, να βελτιώνουν τις διαδικασίες συντήρησης ή μαγειρέματος κ.ά.
- **Αυτόματοι πωλητές.** Οι αυτόματοι πωλητές έχουν ενσωματωμένους επεξεργαστές για να διεκπεραιώνουν τη λειτουργία της πώλησης αγαθών.
- **Εκτυπωτές/Φαξ.** Οι συσκευές αυτές έχουν ενσωματωμένα συστήματα για τον έλεγχο της εκτύπωσης, της μετατροπής των δεδομένων σε εντολές προς τους μηχανισμούς εκτύπωσης και της διασύνδεσης με άλλα συστήματα.
- **Φωτογραφικές μηχανές/Μηχανές λήψης βίντεο.** Και αυτές οι συσκευές διαθέτουν ένα ή παραπάνω ενσωματωμένα συστήματα για την αυτόματη ρύθμιση κάποιων χαρακτηριστικών, ή για την αποθήκευση με βέλτιστο τρόπο των δεδομένων που λαμβάνουν.

1.3.2 Ορισμός του ενσωματωμένου συστήματος

Σύμφωνα με όσα προαναφέρθηκαν, το χαρακτηριστικό των ενσωματωμένων συστημάτων που αναφέρεται στο γεγονός ότι αποτελούνται από υλικό υψηλών επιδόσεων σε συνδυασμό με λογισμικό εξειδικευμένης λειτουργίας, αποτελεί τον βασικό παράγοντα της ανάπτυξής τους, καθώς το τελικό προϊόν απαιτεί πολύ μικρό χρόνο σχεδιασμού και εισόδου στην αγορά (shorttimetomarket) σε σχέση με υπολογιστές γενικού σκοπού και πολλαπλών χρήσεων.

Τα παραδείγματα της προηγούμενης ενότητας αποτελούν ένα μικρό δείγμα των χρησιμοποιούμενων ενσωματωμένων συστημάτων. Η λέξη ενσωματωμένο υποδηλώνει ότι το σύστημα είναι «ενσωματωμένο» σε μια άλλη συσκευή και ότι η λειτουργία του δε θα τροποποιηθεί από τη στιγμή που θα γίνει διαθέσιμο στην αγορά. Από τον ορισμό αυτόν προκύπτει η άποψη ότι το πλήθος των ορισμών για τα ενσωματωμένα συστήματα είναι ίσως ανάλογο του αριθμού των σχεδιαστών που ασχολούνται με αυτά. Έτσι, στο σημείο αυτό προκύπτει το πρόβλημα του ορισμού. Για παράδειγμα, ένας προσωπικός υπολογιστής που εκτελεί μόνο ένα πρόγραμμα (π.χ. κιόσκι πολυμέσων) μπορεί να θεωρηθεί ένα ενσωματωμένο σύστημα. Επίσης, ένας σύγχρονος προσωπικός ψηφιακός βοηθός (PDA) που μπορεί να εκτελέσει ένα πλήθος προγραμμάτων αντίστοιχων με εκείνα του προσωπικού υπολογιστή (π.χ. αποστολή ηλεκτρονικής αλληλογραφίας, ηλεκτρονικός υπολογιστής, επεξεργασία κειμένου, λογιστικών φύλων, παιχνιδιών, αναπαραγωγή πολυμέσων, χρήση διαδικτύου) μπορεί επίσης να θεωρηθεί ένα ενσωματωμένο σύστημα, το οποίο όμως δεν ακολουθεί πιστά τον παραπάνω ορισμό.

Ίσως ένας πιο αυστηρός ορισμός είναι ότι τα ενσωματωμένα συστήματα αποτελούν εξειδικευμένα συστήματα υπολογιστών, τα οποία είναι αφοσιωμένα

στην εκτέλεση μιας συγκεκριμένης λειτουργίας και είναι συνδυασμός υλικού και λογισμικού μέρους. Είναι κοινή πρακτική για τα ενσωματωμένα συστήματα η τοποθέτηση πάνω στην ίδια επιφάνεια ημιαγωγού πυριτίου επεξεργαστικών πυρήνων, μνημών, περιφερειακών, διασυνδέσεων εισόδου/εξόδου και μερικές φορές αναλογικών/ψηφιακών κυκλωμάτων, καταλήγοντας σε ολόκληρα συστήματα πάνω στο ίδιο υλικό πυριτίου (System-on-Chip, SOC). Συνήθως, τα ενσωματωμένα συστήματα αποτελούν τμήμα ενός μεγαλύτερου συστήματος-προϊόντος.

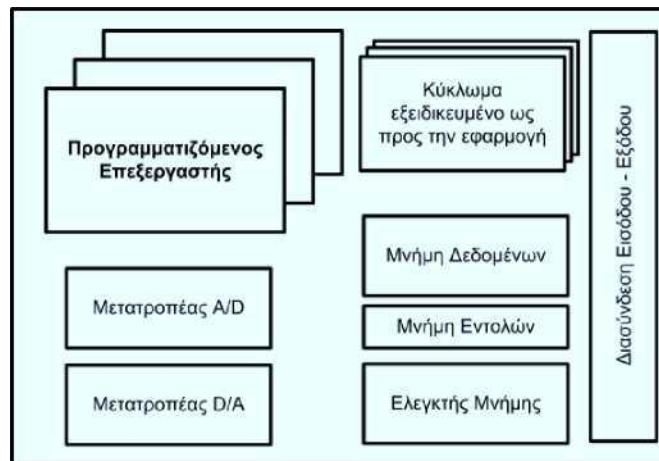
Συνοψίζοντας τα παραπάνω, μπορούν να διακριθούν ορισμένα σαφή χαρακτηριστικά σχετικά με τα ενσωματωμένα συστήματα ως εξής:

- Επιτελούν μια συγκεκριμένη λειτουργία. Σχεδιάζονται για μια μόνο λειτουργία και την εκτελούν αδιαλείπτως (ονομάζονται και «αποκλειστικά» συστήματα).
- Δεν μπορούν να χρησιμοποιηθούν για λειτουργίες διαφορετικές από αυτές για τις οποίες έχουν σχεδιαστεί (μη επαναπρογραμματιζόμενα). Για παράδειγμα, ο ενσωματωμένος επεξεργαστής ενός έξυπνου πλυντηρίου, δε θα μπορούσε να χρησιμοποιηθεί για την αναπαραγωγή πολυμέσων.
- Έχουν αυξημένους περιορισμούς, χάριν επίτευξης όσο το δυνατόν μικρότερου κόστους σχεδιασμού,(κατασκευάζονται σε μεγάλες ποσότητες και με χαμηλό κόστος, ώστε να βρίσκουν ανταπόκριση από το καταναλωτικό κοινό), υψηλών επιδόσεων (ώστε να ανταποκρίνονται αμέσως στις εντολές του χρήστη και να παρουσιάζουν μεγάλη διαδραστικότητα με αυτόν), μικρού μεγέθους (συχνά βρίσκονται σε φορητές συσκευές), χαμηλής κατανάλωσης ενέργειας (διαρκής λειτουργία αλλά μειωμένη απαίτηση συστήματος ψύξης), υψηλής αξιοπιστίας (απρόσκοπτη λειτουργία).

- Βασίζονται σε ένα πλήθος επεξεργαστών, διαφορετικών δομών και αρχιτεκτονικών. Για παράδειγμα, ένας προσωπικός υπολογιστής συνήθως αποτελείται από επεξεργαστές της είτε μόνο της εταιρείας AMD ή της μόνο της INTEL, εν αντιθέσει με τα ενσωματωμένα συστήματα, τα οποία συναντώνται με ένα πλήθος επεξεργαστών.
- Διαθέτουν λειτουργικό πραγματικού χρόνου (Real Time Operating System - RTOS). Σε αντίθεση με τα κλασσικά λειτουργικά συστήματα (Windows, Linux, FreeBSD) που συναντώνται σε προσωπικούς υπολογιστές, τα λειτουργικά συστήματα των ενσωματωμένων συστημάτων λαμβάνουν πάντα σοβαρά υπόψη τις προτεραιότητες των εφαρμογών, και αποδίδουν ολόκληρη την επεξεργαστική ισχύ τους σε εφαρμογές υψηλής προτεραιότητας. Η ευθύνη απόδοσης ή αφαίρεσης προτεραιότητας στις εφαρμογές του συστήματος είναι του σχεδιαστή και είναι υψίστης σημασίας για την σταθερότητα και αξιοπιστία των Ε.Σ., σε σύγκριση με τα λειτουργικά των προσωπικών υπολογιστών.
- Διαθέτουν λειτουργικό σύστημα πολύ μικρού μεγέθους (μερικώνKB), γεγονός που τα καθιστά μη φιλικά προς τον χρήστη (nonuserfriendly) ή τον προγραμματιστή.
- Το λειτουργικό τους σύστημα και οι εφαρμογές τους βρίσκονται τοποθετημένα σε μνήμη ROM (ονομάζεται και μνήμη εντολών). Αντιθέτως, το λειτουργικό σύστημα ενός προσωπικού υπολογιστή ή ενός διακομιστή βρίσκεται σε κάποια αποθηκευτική μονάδα δίσκου (λόγω του μεγάλου μεγέθους που έχει) και είναι εύκολα τροποποιήσιμο. Για να αλλάξει το λογισμικό του ενσωματωμένου συστήματος, πρέπει να αντικατασταθεί η μνήμη ROM με μια άλλη που περιλαμβάνει το νέο πρόγραμμα, μια διαδικασία που απαιτεί εξειδικευμένο προσωπικό. Ωστόσο, τα τελευταία χρόνια τροποποιείται ο έλεγχος της μνήμης, με πολλά Ε.Σ. να έχουν αντί

μνήμης ROM μνήμη FLASH ή EEPROM, η οποία τροποποιείται και γι' αυτό επιτρέπει την εύκολη αναβάθμιση του συστήματος σε περίπτωση ανάγκης.

Ένα ενσωματωμένο σύστημα αποτελείται συχνά από πολλά τμήματα. Στην Εικ.1-2 παρουσιάζονται τα δομικά στοιχεία που συνθέτουν ένα τυπικό ενσωματωμένο σύστημα.



Εικόνα 1-2 Δομικά στοιχεία τυπικού ενσωματωμένου συστήματος [1]

Συνήθως, ένα ενσωματωμένο σύστημα αποτελείται από τα εξής επιμέρους στοιχεία:

- Έναν ή περισσότερους επεξεργαστές (η μνήμη εντολών του επεξεργαστή βρίσκεται εκτός ολοκληρωμένου κυκλώματος, επειδή μπορεί να χρειαστεί να γίνει κάποια μικρή τροποποίηση της λειτουργίας του)
- Μετατροπείς αναλογικού σήματος σε ψηφιακό (A/D converters)
- Μετατροπείς ψηφιακού σήματος σε αναλογικό (D/A converters)
- Μνήμη εντολών του λογισμικού του συστήματος
- Μνήμη δεδομένων όπου αποθηκεύονται ή διαβάζονται τα δεδομένα(αποτελείται συνήθως από μνήμη πάνω στο ολοκληρωμένο

κύκλωμα και από κάποια παρασκηνιακή μνήμη)

- Κυκλώματα εξειδικευμένα για την εφαρμογή που εκτελείται
- Ελεγκτή της μνήμης δεδομένων
- Διασύνδεση εισόδου/εξόδου.

Από τα πιο σημαντικά δομικά στοιχεία που έχει ένα ενσωματωμένο σύστημα είναι ο μικροεπεξεργαστής και η μνήμη, καθώς ο σχεδιασμός και η χρήση αυτών επηρεάζουν σημαντικά την επίδοση και την κατανάλωση ενέργειας του συστήματος.

Η εξέλιξη των μικροεπεξεργαστών τα τελευταία χρόνια είναι σημαντική, καθότι, ενώ αρχικά αποτελούνταν από μερικές χιλιάδες τρανζίστορ, τώρα διαθέτουν εκατομμύρια τρανζίστορ πάνω στο ίδιο ολοκληρωμένο κύκλωμα. Η αλματώδης αυτή πρόοδος που επιτελέστηκε στους επεξεργαστές, κατέστησε προσιτά στο καταναλωτικό κοινό (και με χαμηλό κόστος) τα ενσωματωμένα συστήματα. Η πολυπλοκότητα των παραπάνω επεξεργαστών, σε συνδυασμό με μονάδες μνήμης που περιέχουν πάνω από 256 εκατομμύρια στοιχεία στο ίδιο ολοκληρωμένο κύκλωμα, είναι ο παράγον που έχει καταστήσει δυνατό τον σχεδιασμό επιμέρους συστημάτων ως ένα ολοκληρωμένο κύκλωμα (System On Chip, SOC). Όπως προαναφέρθηκε, ο σχεδιασμός ενός ολοκληρωμένου κυκλώματος διαφοροποιείται σημαντικά από τον σχεδιασμό ενός ASIC (Application Specific Integrated Circuit).

Το δεύτερο σημαντικό δομικό στοιχείο ενός ενσωματωμένου συστήματος, η μνήμη εντός και εκτός του ολοκληρωμένου κυκλώματος, δεν ακολουθεί τους ίδιους ρυθμούς ανάπτυξης με τους επεξεργαστές. Το αποτέλεσμα είναι η δημιουργία ενός ολοένα αυξανόμενου χάσματος ανάμεσα στις συχνότητες λειτουργίας του επεξεργαστή και της μνήμης, γεγονός που δημιουργεί

προβλήματα, ειδικά όταν πρόκειται για εφαρμογές οι οποίες έχουν αυξημένες απαιτήσεις ταχύτητας πρόσβασης στη μνήμη.

Η ανάπτυξη των ενσωματωμένων συστημάτων οδήγησε στην ανάπτυξη της πολυπλοκότητας των εφαρμογών που εκτελούνται από αυτά, σε σημείο να είναι αδύνατον κάποιες εφαρμογές να εκτελεστούν σε πραγματικό χρόνο με τη χρήση των σημερινών ενσωματωμένων συστημάτων. Παρότι δεν είναι ξεκάθαρο εάν η τεχνολογία είναι αυτή που προώθησε την ανάπτυξη τόσο πολύπλοκων εφαρμογών ή οι ίδιες οι επεξεργαστικές απαιτήσεις των εφαρμογών οδήγησαν τους μηχανικούς στον σχεδιασμό καλύτερων συστημάτων, το μόνο σίγουρο είναι πως πρόκειται για ένα πεδίο που προσφέρεται για έρευνα και καινοτομία.

1.4 Εισαγωγή στην αγορά του ενσωματωμένου συστήματος

1.4.1 Κοστολόγηση [1]

Όταν πρόκειται για τον σχεδιασμό Ε.Σ. συστημάτων που προορίζονται για πώληση ως προϊόντα στη βιομηχανία ή ευρύτερα στην αγορά, δε μπορεί να μην έχει προηγηθεί η κοστολόγησή του. Έτσι, το τελικό κόστος του συστήματος προκύπτει ως άθροισμα του μη επαναλαμβανόμενου μηχανικού κόστους (NRE, nonrecurrentengineeringcost,) και του κόστους μονάδας (UC, unitcost).

Το κόστος έρευνας και ανάπτυξης (NRE, nonrecurrentengineeringcost) είναι στην ουσία το κόστος κεφαλαίου και περιλαμβάνει το κόστος που έχει δαπανηθεί από την εταιρία για την ανάπτυξη του συγκεκριμένου προϊόντος σε επίπεδο πληρωμών

προσωπικού, σε σεμινάρια μηχανικών, σε αγορά ποικιλίας υλικών και δημιουργίας πρωτοτύπων, σε δοκιμές ποιότητας ή ηλεκτρομαγνητικής συμβατότητας, σε λειτουργικά έξοδα (φως, νερό, θέρμανση, αναλώσιμα), σε ειδικό εξοπλισμό που απαιτήθηκε για τη δημιουργία του προϊόντος (π.χ. κάποιο εξειδικευμένο παλμογράφο), την αγορά δημοσιεύσεων ή ευρεσιτεχνιών κ.ά. Το κόστος αυτό προϋπολογίζεται και δαπανάται μια φορά για τον σχεδιασμό και την υλοποίηση ενός συγκεκριμένου προϊόντος. Αφού ληφθούν υπόψη τα παραπάνω και ξεκινήσει η μαζική κατασκευή του προϊόντος, τότε μπορεί να υπολογιστεί και η τιμή χονδρικής ή λιανικής πώλησής του.

Το UCαπ' την άλλη είναι ένα επαναλαμβανόμενο κόστος και εμπεριέχει όλα τα επιμέρους κόστη για την αγορά των υλικών, την αποστολή στα κατάλληλα εργοστάσια, το τύπωμα των πλακετών, τη συναρμολόγηση, τη δοκιμή του κάθε προϊόντος, τη μισθοδοσία όσων ασχολούνται με την κατασκευή (ή σε περίπτωση που έχει γίνει εξωτερική ανάθεση (outsourcing) το κόστος αυτής, το πακετάρισμα και την αποστολή στις κεντρικές αποθήκες, κ.ά. Αυτό το κόστος εμφανίζεται κάθε φορά που κατασκευάζεται ένα τέτοιο προϊόν, συνεπώς δε μπορεί να είναι σταθερό, αλλά μεταβάλλεται κάθε φορά ανάλογα με την ισοτιμία των νομισμάτων (όταν υπάρχουν συναλλαγές με νομίσματα διαφορετικής ισοτιμίας), ανάλογα με τον προμηθευτή (μερικές φορές επιλέγονται πιο οικονομικοί προμηθευτές για να μείωση του κόστους), ή ανάλογα και με την ποσότητα των μονάδων που κατασκευάζονται (όσο αυξάνεται ο αριθμός των προϊόντων που κατασκευάζονται τόσο χαμηλώνει το συνολικό κόστος παραγωγής).

Σύμφωνα με τα παραπάνω, ο υπολογισμός της τιμής πώλησης του προϊόντος υπολογίζεται από την ακόλουθη εξίσωση:

$$\text{Κόστος} = \frac{NRE + UC}{1 - M} \quad (1.1)$$

, όπου Μ είναι το περιθώριο κέρδους εκφρασμένο ως ποσοστό επί τοις εκατό (%).

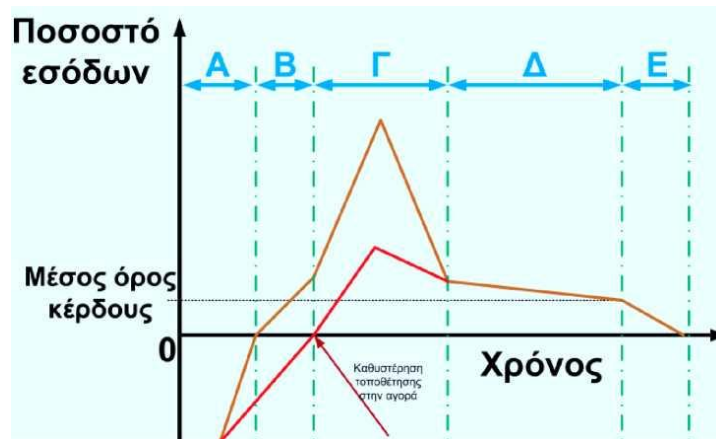
Προκειμένου να μειωθεί το κόστος NRE, συχνά πραγματοποιείται επαναχρησιμοποίηση μεγάλου μέρους του σχεδιασμού του αρχικού προϊόντος, ώστε το NREκόστος να διαιρεθεί σε πολλά προϊόντα. Έτσι, η ίδια βασική αρχιτεκτονική επαναχρησιμοποιείται σε πολλά προϊόντα, με κάποιες μικρές ενδεχομένως διαφοροποιήσεις ή προσθήκες νέων στοιχείων και κατάλληλου κώδικα. Στο σημείο αυτό μπορεί να αναφερθεί ξανά το παράδειγμα της επαγωγικής κουζίνας που αναλύθηκε σε προηγούμενη ενότητα. Όπως αναφέρθηκε, ορισμένα μοντέλα επαγωγικής κουζίνας λειτουργούν χωρίς ένδειξη ώρας ενώ άλλα παρέχουν τη δυνατότητα εμφάνισης της ώρας σε μια οθόνη και χρονοπρογραμματισμό μαγειρέματος. Ωστόσο, και στις δυο περιπτώσεις πρόκειται για την ίδια επαγωγική κουζίνα, με μόνη διαφορά την προσθήκη στον βασικό σχεδιασμό ενός συστήματος RTC με κάποιο κόστος, πέντε νέων οθονών 7 στοιχείων για την εμφάνιση της ώρας, δυο-τριών πλήκτρων και του κατάλληλο λογισμικού. Με τον τρόπο αυτόν, επιτυγχάνεται η διαίρεση του NREκόστους σε 2 προϊόντα, συνεπώς το κόστος ανά προϊόν ελαττώνεται.

1.4.2 Εισαγωγή του προϊόντος στην αγορά

Τη διαδικασία του σχεδιασμού και της κοστολόγησης του προϊόντος, ακολουθεί η εισαγωγή του στην αγορά. Το χρονικό διάστημα από τη σύλληψη της ιδέας ενός προϊόντος ως την προώθησή του στην αγορά περιγράφεται από τον όρο «χρόνος εισαγωγής του προϊόντος» (time-to-market). Αυτό το διάστημα καλύπτει όλες τις φάσεις της έρευνας, της μοντελοποίησης, του σχεδιασμού και της κατασκευής που

αναλύθηκαν στις προηγούμενες παραγράφους, και προκειμένου να μην υπάρχει σημαντική απώλεια εσόδων, πρέπει να είναι αρκετά μικρό.

Στο Σχ.1-5 παρουσιάζεται ενδεικτικό διάγραμμα με όλες τις φάσεις κερδοφορίας από τις οποίες διέρχεται ένα Ε.Σ. ως προϊόν εμπορίου.



Εικόνα 1-5 Διάγραμμα των φάσεων κερδοφορίας από τις οποίες διέρχεται ένα Ε.Σ.[1]

Στον κατακόρυφο άξονα του παραπάνω σχήματος τοποθετείται το ποσοστό των εσόδων, ενώ στον οριζόντιο άξονα τοποθετείται ο χρόνος. Βασικό δείκτη των κερδών μιας εταιρίας αποτελεί η τιμή του ποσοστού των εσόδου συναρτήσει του χρόνου. Τα έσοδα μπορούν να υπολογιστούν από το εμβαδόν του γραφήματος που περικλείεται από τη γραμμή του Ε.Σ. ως τον οριζόντιο άξονα, εάν αφαιρεθεί το εμβαδόν του σχήματος κάτω από τον άξονα. Εννοείται ότι όσο μεγαλύτερο είναι το εμβαδόν, τόσο μεγαλύτερα είναι τα έσοδα. Επιπλέον, η εξάρτηση του ποσοστού εσόδων από τον χρόνο υποδηλώνει πως η καθυστέρηση τοποθέτησης στην αγορά του προϊόντος οδηγεί σε μείωση του κέρδους.

Βάσει του παραπάνω διαγράμματος, οι φάσεις από τις οποίες διέρχεται ένα προϊόν είναι οι ακόλουθες: Μια σωστή τοποθέτηση του προϊόντος ακολουθεί την πορτοκαλί γραμμή (που έχει και τα περισσότερα έσοδα).

- Στην πρώτη φάση (φάση Α) δεν έχει αναπτυχθεί το προϊόν και έτσι η εταιρία δεν έχει έσοδα παρά μόνο έξοδα. Τα έξοδα αυτά κυρίως είναι τα ΝREπου προαναφέρθηκαν. Στην ουσία, πρόκειται για τη φάση της έρευνας και ανάπτυξης (R&D, Research&Development).
- Στη δεύτερη φάση (φάση Β), το Ε.Σ. προωθείται ως προϊόν προς πώληση. Συνεπώς στη φάση αυτή επιτυγχάνονται έσοδα. Σε διαφορετική περίπτωση - αν δηλαδή σε αυτό το σημείο δεν υπάρχουν έσοδα-υπάρχει ένδειξη ότι το προϊόν δεν έχει αποδοχή από το αγοραστικό κοινό και χρίζει κατάλληλης τροποποίησης.
- Στην φάση που ακολουθεί, τη φάση Γ, εντοπίζεται μια μερίδα καταναλωτών, οι οποίοι ονομάζονται «πρώιμοι καταναλωτές»(early adopters). Όταν συγκεντρωθεί μια κρίσιμη μάζα αυτών των χρηστών και πουληθεί ένας ικανοποιητικός αριθμός Ε.Σ., τότε το προϊόν προωθείται αυτόματα μέσω κοινωνικών ιστότοπων (forum) και περιοδικών τεχνολογικού περιεχομένου, με σκοπό την εκτόξευση των πωλήσεων. Ωστόσο, όπως προκύπτει και από το διάγραμμα, κάποια στιγμή τα έσοδα αρχίζουν να μειώνονται και αυτό είτε επειδή το προϊόν το έχει ήδη αγοράσει από αρκετούς καταναλωτές είτε επειδή, λόγω ανταγωνισμού, άλλες εταιρίες έχουν προωθήσει παρόμοια προϊόντα.
- Περνώντας στη φάση Δ, ο ρυθμός μείωσης των εσόδων ελαττώνεται. Το Ε.Σ. βρίσκεται ακόμη προς πώληση, αλλά οι πωλήσεις του είναι σταθερές.

Κάποια στιγμή, το Ε.Σ. θα αποσυρθεί από την αγορά, γιατί δε θα καλύπτει τις νέες ανάγκες, και η κατασκευάστρια εταιρία δε θα μπορεί να συντηρεί και να λειτουργεί μια γραμμή παραγωγής για ένα παρωχημένο προϊόν.

- Η φάση Ε περιλαμβάνει την απόσυρση του Ε.Σ. και ονομάζεται τέλος ζωής (EOL, End-Of-Life).

Βάσει του διαγράμματος, αν ένα προϊόν καθυστερήσει να εισαχθεί στην αγορά (κόκκινη γραμμή), τότε είναι φανερό ότι τα έσοδα είναι αρκετά λιγότερα. Για τον λόγο αυτόν υπάρχουν συγκεκριμένες προδιαγραφές για τον χρόνο εισαγωγής του προϊόντος στην αγορά (ενδεικτικά 6 μήνες), όπως επίσης και τυπική διάρκεια ζωής του σε αυτήν (ενδεικτικά 18 μήνες).

1.4.3 Εμπορικές και κατασκευαστικές προδιαγραφές των Ε.Σ.

Σύμφωνα με όσα αναφέρθηκαν σε προηγούμενες ενότητες αναφορικά με τα διαφορετικά χαρακτηριστικά, τη δομή και την αρχιτεκτονική διαφορετικών Ε.Σ. (όπως το φάσμα διακύμανσης από 8bit επεξεργαστές έως και 32bit, από διάφορες εταιρίες, με πλήθος λειτουργικών συστημάτων, κ.ά.), εξάγεται το συμπέρασμα ότι υπάρχει αρκετή ανομοιογένεια και ποικιλία συστημάτων, ώστε να καλύπτονται οι απαιτήσεις διαφορετικών εφαρμογών. Στο σημείο αυτό, κλείνοντας το εισαγωγικό κεφάλαιο, συγκεντρώνονται και συνοψίζονται ορισμένα χαρακτηριστικά που έχουν αναφερθεί και σε προηγούμενες ενότητες, κοινά στα περισσότερα Ε.Σ.:

- Εκτελούν μια καθορισμένη λειτουργία ή αρκετά περιορισμένο αριθμό διεργασιών. Όλα τα Ε.Σ. εκτελούν ατέρμων βρόχο προγράμματος που λαμ-

βάνει δεδομένα από αισθητήρια και επιτελεί κάποιες ενέργειες.

- Έχουν περιορισμένες δυνατότητες. Χρησιμοποιούν πολύ λιγότερους πόρους (π.χ. επεξεργαστικούς ή μνήμης) ως προς τα αντίστοιχα υπολογιστικά συστήματα των επιτραπέζιων υπολογιστών, και αυτό επειδή ο κατασκευαστής ενός ΕΣ γνωρίζει ακριβώς τους πόρους που χρειάζεται η εφαρμογή του και επιλέγει το υλικό που ταιριάζει καλύτερα.
- Έχουν χαμηλή κατανάλωση ενέργειας και ισχύος. Αξίζει εδώ να σημειωθεί η διαφορά μεταξύ της κατανάλωσης ισχύος και της κατανάλωσης ενέργειας. Η κατανάλωση ισχύος αναφέρεται στο έργο που καταναλώνεται στη μονάδα του χρόνου ($\text{Joule/second}=\text{Watt}$ στο σύστημα S.I) ενώ η ενέργεια που καταναλώνεται σε ένα σύστημα μετράται σε KWatt*hour ή Joules, και αφορά τα την ισχύ που καταναλώθηκε σε όλο το χρονικό διάστημα. Για παράδειγμα, ένα Ε.Σ. που καταναλώνει ισχύ 1 Wattσε 100 ώρες έχει συνολική κατανάλωση ενέργειας 100 Watt*hours. Ένα άλλο Ε.Σ. που καταναλώνει ισχύ 100 Wattσε 1 ώρα, έχει ίδια συνολική κατανάλωση ενέργειας 100 Watt*hours, αλλά χαμηλότερη κατανάλωση ισχύος, συνεπώς είναι καλύτερο.
- Είναι αντιδραστικά και λειτουργούν σε πραγματικό χρόνο. Αρκετά Ε.Σ. χρησιμοποιούνται σε εφαρμογές που απαιτούν την τήρηση αυστηρών χρονικών περιορισμών. Για αυτόν τον λόγο απαιτούνται επεξεργαστές με προβλέσιμο χρόνο εξυπηρέτησης διακοπών και λειτουργικά συστήματα που έχουν δημιουργηθεί ειδικά για ανταπόκριση σε πραγματικό χρόνο (λειτουργικά συστήματα πραγματικού χρόνου, RTOS, Real Time Operating Systems).
- Είναι αξιόπιστα. Τα Ε.Σ. κατασκευάζονται για να λειτουργούν αδιάκοπα,

χωρίς την παρέμβαση κάποιου χειριστή. Επιπλέον, μπορούν να τοποθετηθούν σε πολύ μεγάλα υψόμετρα (π.χ. δορυφόροι) ή σε πολύ χαμηλά (υποβρύχια αισθητήρια), μακριά από ανθρώπινη πρόσβαση.

- Έχουν Μικρό κόστος. Το κόστος παραγωγής στα Ε.Σ. είναι μικρό, αφού κατασκευάζονται σε πολύ μεγαλύτερες ποσότητες από ό,τι τα υπόλοιπα υπολογιστικά συστήματα. Για αυτόν τον λόγο στα Ε.Σ., οι 8 bitεπεξεργαστές καταλαμβάνουν ένα σημαντικό τμήμα της παγκόσμιας αγοράς επεξεργαστών (περίπου 25%), αν και σε όλα τα προηγμένα υπολογιστικά συστήματα τα 8bitσυστήματα έχουν εξαφανιστεί. Ακόμη και σήμερα κατασκευάζονται και χρησιμοποιούνται ευρέως 8 bitεπεξεργαστές που είχαν σχεδιαστεί κατά τη δεκαετία του 1980, όπως ο Intel8051, λόγω του χαμηλού τους κόστους σε συνδυασμό με τη χαμηλότερη κατανάλωση ενέργειας για την υλοποίηση των ίδιων προδιαγραφών.
- Έχουν μικρό μέγεθος και βάρος. Τα Ε.Σ. είναι ενσωματωμένα μέσα σε άλλα συστήματα και έτσι καταλαμβάνουν όσο το δυνατό μικρότερο όγκο. Αυτό συνεπάγεται λιγότερο υλικό, συνεπώς και πιο περιορισμένους πόρους.
- Διαθέτουν διαρκή λειτουργία και μεγάλη διάρκεια ζωής. Τα Ε.Σ. συνήθως από τη στιγμή που τίθενται σε λειτουργία, λειτουργούν αδιάλειπτα, και μάλιστα για πολύ μεγάλο διάστημα. Σε αντίθεση με έναν επιτραπέζιο υπολογιστή που έχει εκτιμώμενο χρόνο ζωής μερικά χρόνια, τα Ε.Σ. έχουν εκτιμώμενο χρόνο ζωής πάνω από δεκαετία, που σημαίνει ότι σχεδιάζονται κατάλληλα ώστε να «αντέξουν» και να αντιμετωπίσουν με επιτυχία οποιοδήποτε πρόβλημα προκύψει.
- Συναντούν δυσκολία στην ενημέρωση του λογισμικού τους. Σε αντίθεση με τους τυπικούς υπολογιστές που μπορούν και ενημερώνουν το λογισμικό

τους μέσω του διαδικτύου (update), ένα Ε.Σ. κατασκευάζεται χωρίς δυνατότητα ανανέωσης του λογισμικού τους, και έτσι δύσκολα διορθώνονται προβλήματα (κυρίως ασφάλειας) σε διεργασίες που εκτελούνται, και άρα είναι πιο επιρρεπή στη μη εξουσιοδοτημένη πρόσβαση.

- Είναι ασφαλή. Αρκετά Ε.Σ. τοποθετούνται σε κρίσιμες δομές, όπως για τον έλεγχο ενός πυρηνικού αντιδραστήρα. Επιπλέον, προστατεύονται όσο το δυνατόν καλύτερα, αφού αν αποκτήσει κάποιος κακόβουλος χρήστης πρόσβαση σε αυτά θα μπορέσει να τα απενεργοποιήσει ή να τα καταστρέψει. Η ιστορία έχει καταγράψει πραγματικά γεγονότα που αφορούν τη δημιουργία ιών που κατέστρεψαν τα Ε.Σ. σε κρίσιμα συστήματα, όπως οι πυρηνικοί αντιδραστήρες του Ιράν, καθώς και τον απομακρυσμένο έλεγχο των Ε.Σ. ενός οχήματος μέσω δικτύου κινητής τηλεφωνίας από κακόβουλους χρήστες.
- Κατασκευάζονται σχεδόν αυτόματα και σε μεγάλες ποσότητες για να υπάρχει μειωμένο κόστος. Ο όρος κατασκευασιμότητα περιγράφει τις τεχνικές και τα στοιχεία που πρέπει να επιλεγθούν, ώστε να μπορεί εύκολα να κατασκευαστεί σε μεγάλες ποσότητες. Σε αυτό τον όρο συγκαταλέγονται εκτός από τον σχεδιασμό της τυπωμένης πλακέτας και του περιβλήματος (housing) και η επιλογή των προϊόντων και των προμηθευτών, ώστε να μπορούν να παρέχουν τα στοιχεία που απαιτούνται και να μην παρατηρηθεί κάποια έλλειψη. Ένα συναφές παράδειγμα αποτελεί ο επεξεργαστής 8051, ο οποίος κατασκευάζεται από πολλές διαφορετικές εταιρίες είναι ένα προϊόν που αποτελεί μια ασφαλή επιλογή για 8bitεπεξεργαστή, σε σύγκριση με έναν επεξεργαστή 8bitπου κατασκευάζεται από μια μονό εταιρία, η οποία μπορεί να χρεοκοπήσει.

- Έχουν μικρό χρόνο εισαγωγής τους στην αγορά. Τα Ε.Σ. προορίζονται κυρίως για καταναλωτικές συσκευές, οπότε έχουν πολύ σύντομο χρόνο ολοκλήρωσης. Διαφορετικά, ο κατασκευαστής ενδέχεται να ζημιωθεί, όπως έχει συζητηθεί προηγουμένως.

Όπως φαίνεται από τις παραπάνω προδιαγραφές ,ορισμένες απαιτήσεις χαρακτηριστικών βρίσκονται σε αντιπαράθεση μεταξύ τους. Παραδείγματος χάριν, η επίτευξη της αξιοπιστίας έρχεται σε σύγκρουση με το μειωμένο χρόνο σχεδίασης. Δικαιολογημένα λοιπόν, ο σχεδιασμός των Ε.Σ. θεωρείται δύσκολος, με απαιτήσεις εξειδικευμένων γνώσεων σε πολλούς τομείς, και για αυτό προτιμώνται μηχανικοί σχεδιασμού Ε.Σ. με αρκετά μεγάλη εμπειρία και δεξιότητες.

ΚΕΦΑΛΑΙΟ 2: ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΕΝΣΩΜΑΤΩΜΕΝΩΝ ΣΥΣΤΗΜΑΤΩΝ [6]

2.1 Εισαγωγή

Κατά τον σχεδιασμό ενός Ε.Σ., λαμβάνονται υπόψη από τον σχεδιαστή οι απαιτήσεις της εφαρμογής, τα χαρακτηριστικά της διαθέσιμης αρχιτεκτονικής και ο σκοπός χρήσης του εκάστοτε συστήματος, βάσει των οποίων προτείνονται και οι κατάλληλες λύσεις. Από πλευράς σχεδιαστή, απαιτείται διαρκής ανάλυση και να εκτίμηση πιθανών συμβιβασμών ή και βελτιστοποιήσεων, που να αφορούν σε συγκεκριμένα χαρακτηριστικά της εφαρμογής του.

Κατά το σύννηδες, βασική απαίτηση από τον σχεδιαστή αποτελεί η μεγιστοποίηση της επίδοσης του συστήματος, με όσο το δυνατό μικρότερο κόστους κάθε πτυχής του. Αναπόφευκτα, διαπιστώνεται πληθώρα λύσεων που καλύπτουν την παραπάνω απαίτηση, κάθε μια από τις οποίες ικανοποιεί με διαφορετικό τρόπο, ανάλογα με τις εκάστοτε ανάγκες της εφαρμογής για την οποία προορίζεται. Για παράδειγμα, ο σχεδιασμός ενός ψηφιακού ρολογιού απαιτεί ελαχιστοποίηση του κόστους, με ταυτόχρονη κάλυψη των ελάχιστων απαιτούμενων επεξεργαστικών και λειτουργικών απαιτήσεων. Ένα διαφορετικό παράδειγμα αποτελεί η περίπτωση στην οποία ζητείται η σχεδίαση ενός ψηφιακού προσωπικού βοηθού(PDA, Personal Digital Assistant), όπου η προσπάθεια μεγιστοποίησης των επιδόσεων της συσκευής συγκριτικά με όμοιές τις συσκευές επιβάλλει μικρότερο συμβιβασμό στο κομμάτι του κόστους της συσκευής.

Συμπερασματικά, βασική βαθμίδα για την επίτευξη του στόχου κάθε σχεδιασμού,

που δεν είναι άλλος από τη μεγιστοποίηση της επίδοσης με όσο το δυνατόν μικρότερο κόστος, αποτελεί η μοντελοποίηση του συστήματος, η οποία αναλύεται στη συνέχεια του κεφαλαίου.

2.2 Αναγκαιότητα της μοντελοποίησης

2.2.1 Σημασία της μοντελοποίησης

Η μοντελοποίηση είναι ένα σύνολο τεχνικών που προηγείται της υλοποίησης ενός συστήματος, επιτρέποντας στον σχεδιαστή να επιτύχει τον στόχο του με την περισσότερη δυνατή ακρίβεια, έχοντας κάνει τις απαραίτητες προβλέψεις απόκρισης του συστήματός του. Παρόλη την εξέχουσα σημασία της, παρατηρείται συχνά, λόγω έλλειψης χρονικού περιθωρίου, ανάπτυξη ελλιπούς μοντέλου ή και καθολική παράλειψη μοντελοποίησης του συστήματος που πρόκειται να κατασκευαστεί. Στις περιπτώσεις αυτές, δημιουργείται αβάσιμα η πεποίθηση της επίτευξης του προϊόντος ώστε να λειτουργεί με τον επιθυμητό τρόπο, κατόπιν σύνδεσης των δομικών στοιχείων και ανάπτυξης των εντολών κώδικα. Ωστόσο, κάτω από αυτές τις συνθήκες, ο κίνδυνος αστοχίας είναι αυξημένος και μπορεί να οδηγήσει στο να παρέλθει η διορία εισαγωγής του προϊόντος στην αγορά. Οι οικονομικές συνέπειες μπορούν να αποβούν μοιραίες σε μια τέτοια περίπτωση. Επιπλέον, στην ευχάριστη περίπτωση κατά την οποία το προϊόν κατασκευάζεται και λειτουργεί σύμφωνα με τις απαιτήσεις, ο παράγοντας να αποτύχει τελικά κατά τη λειτουργία υπό μη αποδεκτές συνθήκες είναι πάλι αυξημένος, εφόσον δεν έχει προηγηθεί έλεγχος του μοντέλου, λόγου χάρη σε κάποιο περιβάλλον εξομοίωσης, ώστε να δοθούν τα περιθώρια και η ανοχή του συστήματος υπό διαφορετικές συνθήκες λειτουργίας.

Πληθώρα παραδειγμάτων που αποδεικνύουν τη σημασία της μοντελοποίησης μπορούν να βρεθούν σε κάθε χώρο ανάπτυξης με συστήματα όπως αυτά που εξετάζονται στην εν λόγω πτυχιακή εργασία. Ορισμένα τέτοια θα δοθούν στη συνέχεια.

Η μοντελοποίηση αποτελεί απαραίτητη προϋπόθεση κατά την ανάπτυξη ενός συστήματος και η σημασία της αυξάνεται όσο αυξάνονται η πολυπλοκότητα και οι απαιτήσεις του συστήματος. Αν δεν προηγηθεί μοντελοποίηση, το τελικό αποτέλεσμα αμφισβητείται και πιθανότατα οδηγεί σε υπερβολική εργασία του σχεδιαστή προκειμένου να διορθώσει τα προβλήματα τα οποία εμφανίζονται. Έτσι, διαπιστώνεται συχνά υπέρμετρη ενασχόληση των σχεδιαστών με ένα συγκεκριμένο προϊόν, γεγονός που δεν τους αφήνει χρονικά περιθώρια για την ανάπτυξη άλλων εφαρμογών ή, ακόμα χειρότερα, δύναται να οδηγήσει στο να χαθεί κάποια κρίσιμη ημερομηνία παράδοσης λόγω σφαλμάτων που μπορούν να προκύψουν και να απαιτούν ανακατασκευή του προϊόντος. Ασφαλώς, όπως η παντελής έλλειψη μοντελοποίησης των προϊόντων προς υλοποίηση που αναφέρθηκε και νωρίτερα αποτελεί την πιο επικίνδυνη επιλογή. Ωστόσο, μια ελλιπής μοντελοποίηση ή λάθος εκτιμήσεις μπορούν να οδηγούν στο ίδιο αποτέλεσμα, οπότε η μοντελοποίηση δεν πρέπει απλώς να γίνεται, αλλά να γίνεται και σωστά. Με λίγα λόγια, ο χρόνος που δαπανάται κατά τη σχεδίαση ενός προϊόντος αποδεικνύεται στην πράξη βασική προϋπόθεση για να μπορούν να παρέχονται οι απαραίτητες εγγυήσεις στο αγοραστικό κοινό. Άλλωστε, δεν είναι τυχαίο το γεγονός ότι οι πολυεθνικές εταιρίες δαπανούν μεγάλα χρηματικά ποσά για εκπαίδευση και εξοπλισμό συστημάτων μοντελοποίησης (προγράμματα εξομοίωσης, αισθητήρια, όργανα μέτρησης κ.ά.).

Η μοντελοποίηση, που αποδεδειγμένα αποτελεί απαραίτητο εργαλείο των μηχανικών, δεν αφορά μόνο τον κατασκευαστικό τομέα, αλλά χρησιμοποιείται παντού: στον κατασκευαστικό τομέα και τον τομέα των μέσων μαζικής

(αεροσκάφη, πλοία, τρένα κ.ά.) και ιδιωτικής μεταφοράς (αυτοκίνητα οχήματα), σε δορυφορικές εφαρμογές, στην ανάπτυξη καινοτόμων επεξεργαστικών μονάδων και συστοιχιών, στη μοντελοποίηση ενός τοπικού δικτύου καθώς και του διαδικτύου, και σε ποικίλες άλλες τεχνικές εφαρμογές που περιλαμβάνουν πολύπλοκα εξαρτήματα και λογική ελέγχου, η εξασφάλιση της οποίας χρήζει μοντελοποίησης. Τα μοντέλα που χρησιμοποιούνται κατά κύριο λόγο στα Ε.Σ. είναι αμιγώς μαθηματικά, αλλά δεν αποκλείεται να προκύψει ανάγκη χρησιμοποίησης κάποιου μοντέλου διαφορετικής φύσης (λόγου χάρη ένα μοντέλο απεικόνισης ή επικοινωνίας κ.ά.), τα οποία ασφαλώς χρησιμοποιούν στοιχεία των αμιγώς μαθηματικών μοντέλων.

Ένα γενικό ορισμό του μοντέλου θα μπορούσε να περιλαμβάνει οτιδήποτε μπορεί, μέσα από παραδοχές, να αναπαραστήσει τη λειτουργία ενός συστήματος υπό ποικίλες συνθήκες και αποσκοπεί στην διαπίστωση τυχόν σφαλμάτων και αποκλίσεων από την αναμενόμενη λειτουργία του συστήματος που σχεδιάζεται.

Ένα πλήρες και σωστά ανεπτυγμένο μοντέλο πρέπει να επιτυγχάνει τα εξής:

- Όσο το δυνατό πιστή αναπαράσταση του συστήματος
- Ρεαλιστική προσομοίωσης της συμπεριφοράς του συστήματος υπό διαφορετικές συνθήκες λειτουργίας
- Καθοδήγηση βημάτων προς την ολοκλήρωση του συστήματος
- Διευκόλυνση της δημιουργία πολλαπλών αντιγράφων του ίδιου ή παρόμοιων προϊόντων
- Επιβεβαίωση των προδιαγραφών του συστήματος πριν από το στάδιο της κατασκευής του

- Γρήγορη αναζήτηση του σχεδιασμού που είναι ο επιθυμητός ανάμεσα σε διάφορες παραλλαγές που διατίθενται
- Παροχή πληροφορίας-απεικόνισης του τελικού προϊόντος ώστε να επιτυγχάνεται καλύτερη συνεννόηση μεταξύ των διαφορετικών ατόμων που εμπλέκονται στον σχεδιασμό

Όσο πιο μεγάλη είναι η πολυπλοκότητα σε ένα σύστημα, τόσο πιο αναγκαία είναι η ανάπτυξη ενός μοντέλου για την καλύτερη μελέτη του και την άρση των αλληλεξαρτήσεων από τους μηχανικούς που ασχολούνται με τα επιμέρους τμήματα από τα οποία αποτελείται.

Για την καλύτερη αντιμετώπιση ενός πολύπλοκου συστήματος, συνήθως χρησιμοποιείται μια μέθοδος προσέγγισης σύνθετων προβλημάτων η οποία έχει προταθεί από τον Edsger Dijkstra [7] και καλείται αλγόριθμος του Dijkstra, η οποία προτείνει τη διαίρεση ενός σύνθετου προβλήματος σε πολλά επιμέρους τμήματα μικρότερης πολυπλοκότητας, τα οποία αποσπασματικά μπορούν να επιλυθούν και τελικά να προκύψει η επίλυση του αρχικού προβλήματος μεγάλης πολυπλοκότητας. Ουσιαστικά η παραπάνω μέθοδος σκιαγραφεί τη μοντελοποίηση στην πράξη, καθώς ένα σύστημα αναλύεται σε επιμέρους τμήματά τα οποία αναλύονται και στη συνέχεια συνθέτουν το συνολικό σύστημα.

2.2.2 Οικονομικά στοιχεία μοντελοποίησης

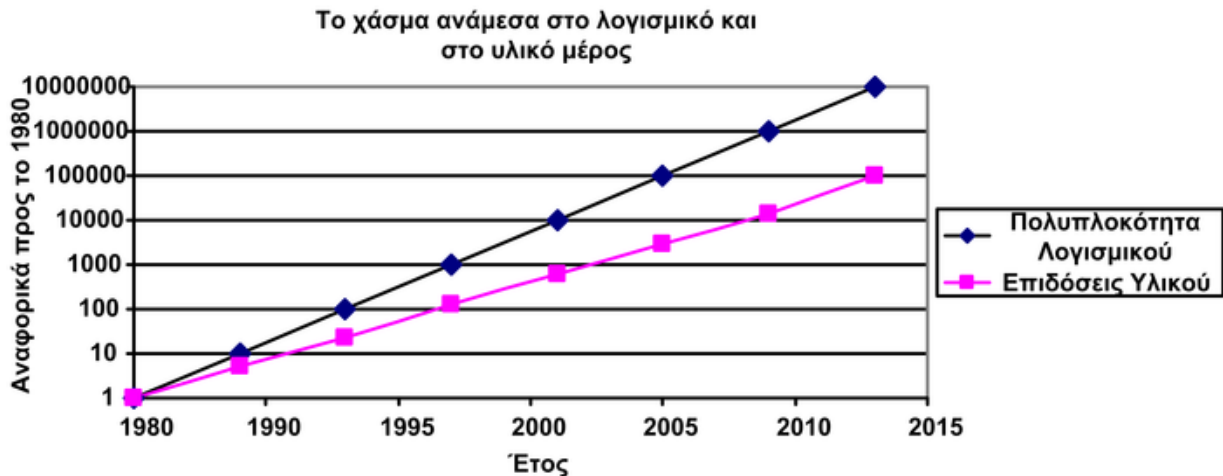
Η ραγδαία ανάπτυξη των ενσωματωμένων συστημάτων, όπως συζητήθηκε σε προηγούμενη ενότητα, οφείλεται στην πρόοδο που επιτεύχθηκε τόσο στο υλικό μέρος (π.χ. επεξεργαστές αυξημένων επιδόσεων) όσο και στο κομμάτι του

λογισμικού. Το λογισμικό είναι ο οργανωτής και εντολοδόχος του υλικού προκειμένου να επιτυγχάνεται σωστά και έγκυρα η προδιαγεγραμμένη λειτουργία του συστήματος. Οι ιδιότητες του υλικού επηρεάζουν σημαντικά την ταχύτητα, την απόκριση και την κατανάλωση ενέργειας του συστήματος (π.χ. στην περίπτωση ενός επεξεργαστή που εργάζεται με μικρές συχνότητες λειτουργίας συνεπάγεται ότι όλο το σύστημα καθυστερεί λόγω αυτού, ανεξάρτητα από το αν διαθέτει πολύ γρηγορότερα στοιχεία, π.χ. διαύλους επικοινωνίας κ.ά.). Γι' αυτό, σε τέτοιες εφαρμογές, το λογισμικό συμβάλλει καταλυτικά στη διαμόρφωση όλων των μετρικών ενός συστήματος, καθώς όσο πιο υψηλή είναι η απόδοση του λογισμικού τόσο λιγότερο μειώνεται και η απόδοση του υλικού. Σε αντίθετη περίπτωση, επιβαρύνονται και τα πιο ανεπτυγμένα υλικά μέρη του Ε.Σ., καθώς ένα σύστημα γενικά είναι τόσο γρήγορο όσο το αργότερο στοιχείο που διαθέτει.

Συνεπώς, το λογισμικό επιδρά σε μεγάλο βαθμό στην απόδοση του υλικού, γι' αυτό και η ταυτόχρονη βελτιστοποίηση και των δύο ταυτόχρονα συνεπάγεται αδιαμφισβήτη βελτίωση του συνολικού συστήματος, ενώ η βελτιστοποίηση του ενός μόνο δεν συνεπάγεται αυτόματη αύξησης του συνολικού συντελεστή απόδοσης. Παρόλο που και τα δύο είναι αναπόσπαστα μέρη του συστήματος, λόγω του ότι η εξέλιξη του υλικού έχει φτάσει σε ένα αρκετά υψηλό επίπεδο και περαιτέρω βελτιώσεις γίνονται με αργούς ρυθμούς, τα τελευταία χρόνια έχει δοθεί μεγαλύτερη έμφαση στο λογισμικό και, ιδιαίτερα, σε αποδοτικότερες μεθόδους μοντελοποίησης και προγραμματισμού των εφαρμογών, ειδικά στον τομέα των Ε.Σ. Αυτό οφείλεται εν μέρει και στην τάση που επικρατεί να χρησιμοποιούνται διακριτά εξαρτήματα υλικού (discrete components).

Από τα παραπάνω προκύπτει ότι το ζήτημα της ανάπτυξης ενός συστήματος επικεντρώνεται περισσότερο στη βελτίωση του λογισμικού, που όπως ειπώθηκε,

κατευθύνει το υλικό. Έτσι, πλέον, η πολυπλοκότητα των εφαρμογών που προορίζονται για τα Ε.Σ. μπορεί να θεωρηθεί ο κυρίαρχος παράγοντας διαμόρφωσης του κόστους. Σε αντίθεση με τις παλαιότερα Ε.Σ. τα οποία περιλάμβαναν μικρής πολυπλοκότητας προγράμματα για απλές λειτουργίες, τα σύγχρονα Ε.Σ. περιλαμβάνουν λειτουργίες οι οποίες δεν είναι δυνατό να υλοποιηθούν με απλό λογισμικό. Με άλλα λόγια, με την απαίτηση της αγοράς για Ε.Σ. με δυνατότητα περισσότερων λειτουργιών δημιουργήθηκε και η απαίτηση για καλύτερες επιδόσεις σε επίπεδο υλικού, για προγράμματα φιλικά προς το χρήστη (userfriendly) και για εφαρμογές αρκετά απαιτητικές (λόγου χάρη η ψηφιακή επεξεργασία εικόνας). Γι' αυτόν τον λόγο δίδεται αρκετά μεγαλύτερη έμφαση στην ανάπτυξη της ποικιλίας και της πολυπλοκότητας των εφαρμογών που καλούνται να επιτελέσουν τα Ε.Σ.



Σχήμα 2-1 Ρυθμός αύξησης της πολυπλοκότητας των εφαρμογών των Ε.Σ. συναρτήσει των επιδόσεων των επεξεργαστών [1]

Σύμφωνα με τα παραπάνω, η τάξη της πολυπλοκότητας των εφαρμογών που αναπτύσσονται αποτυπώνει και το επίπεδο ανάπτυξης των Ε.Σ. Η πολυπλοκότητα μιας εφαρμογής εξαρτάται από τα ακόλουθα:

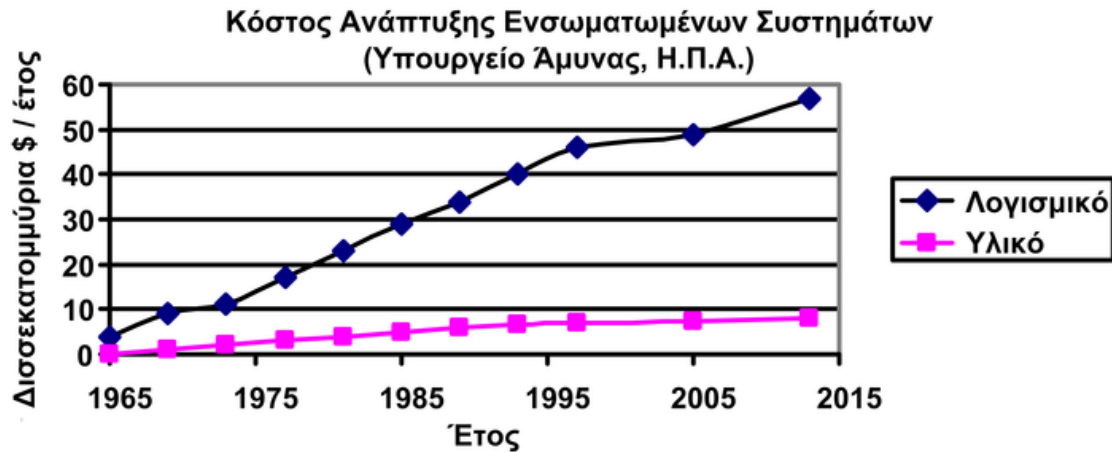
- το πλήθος και την πολυπλοκότητα των εντολών και των συναρτήσεων που περιλαμβάνει,
- τις διαφορετικές λειτουργίες που επιτελεί,
- το μέγεθος κώδικα που αποθηκεύεται στη μνήμη εντολών και
- τη μνήμη που καταλαμβάνει ο όγκος των δεδομένων του λογισμικού.

Όπως φαίνεται από το Σχ.2-1, ο μέσος όρος της πολυπλοκότητας των εφαρμογών ακολουθεί μια εκθετική αύξηση στον χρόνο. Αν στο ίδιο γράφημα τοποθετηθούν και οι επιδόσεις των επεξεργαστών, φαίνεται ότι αυτές υστερούν, μη μπορώντας να καλύψουν τις τρέχουσες απαιτήσεις των εφαρμογών. Διαφαίνεται λοιπόν ένα χάσμα ανάμεσα στις δυνατότητες εκτέλεσης ενός συστήματος (επιδόσεις υλικού) και στις απαιτήσεις των εφαρμογών οι οποίες θα εκτελεστούν (πολυπλοκότητα λογισμικού).

Το πρόβλημα της δημιουργίας λογισμικού για Ε.Σ. είναι βαρύνουσας σημασίας για τις πολυεθνικές εταιρείες που έχουν κυριαρχήσει στον τομέα της ανάπτυξης του λογισμικού των υπολογιστικών συστημάτων, των τηλεπικοινωνιών και των διακομιστών, γι' αυτό και δαπανούν σημαντικό μέρος του προϋπολογισμού τους στην έρευνα και την ανάπτυξη (R&D) του software εφαρμογών για Ε.Σ.

Παράδειγμα που επιβεβαιώνει το γεγονός ότι το λογισμικό των Ε.Σ. είναι ο τομέας

στον οποίο δίδεται περισσότερη σημασία αποτελεί το Υπουργείο Άμυνας των Η.Π.Α., ένας από τους μεγαλύτερους αγοραστές Ε.Σ. παγκοσμίως, για τον οποίο το κόστος ανάπτυξης του λογισμικού είναι έως και 6 φορές μεγαλύτερο από το κόστος ανάπτυξης των υλικών μερών, όπως φαίνεται στο Σχ.2-2.



Σχήμα 2-2 Το λογισμικό κυριαρχεί στη διαμόρφωση του κόστους ενός Ε.Σ. [1]

Επιπρόσθετα, σύμφωνα με τα στοιχεία που της έκθεση του Υπουργείου Εμπορίου και Βιομηχανίας της Ολλανδίας, ο αριθμός των ατόμων ή εταιρειών που υπέγραψαν συμβόλαια για ανάπτυξη κώδικα σε Ε.Σ. αυξήθηκε κατά 11 εκατομμύρια από το 1994 έως το 2015, ενώ ο αριθμός των κατασκευαστών υλικού Ε.Σ. αυξήθηκε κατά 0.5 εκατομμύρια στο ίδιο διάστημα [1].

Το συμπέρασμα που μπορεί να εξαχθεί από τα όσα αναφέρθηκαν παραπάνω είναι ότι ο σωστός σχεδιασμός ενός Ε.Σ. προϋποθέτει πετυχημένη μοντελοποίηση τόσο σε επίπεδο λογισμικού όσο και σε επίπεδο υλικού, προκειμένου να επιτευχθεί η μέγιστη απόδοση ενός Ε.Σ. Ωστόσο, όπως προκύπτει, ο σχεδιασμός των δυο αυτών μερών μπορεί να γίνει είτε ανεξάρτητα είτε παράλληλα, αρκεί να

εφαρμοστεί συνδυαστικά με επιτυχία σε ένα πλήθος εφαρμογών, που εκτελούνται σε Ε.Σ., γεγονός που δείχνει τη γενική χρήση που μπορούν να έχουν.

2.2.3 Εξάρτηση του NRE από τη μοντελοποίηση

Το NRE ενός Ε.Σ. που συζητήθηκε σε προηγούμενο κεφάλαιο είναι ζωτικής σημασίας για τη διαμόρφωση της τιμής του τελικού προϊόντος, αφού ενσωματώνεται στην τελική τιμή. Όσο μεγαλύτερο είναι αυτό το αρχικό κόστος, τόσο περισσότερο αυξάνεται η τιμή των τελικών προϊόντων και ενδεχομένως θα επηρεάζεται αρνητικά ο αριθμός των πωλήσεων, ζημιώνοντας την εταιρία που το προμηθεύει στην αγορά. Το κόστος NRE για το ίδιο προϊόν σε δυο διαφορετικές εταιρίες μπορεί να είναι διαφορετικό. Προκειμένου να ελαχιστοποιηθεί αυτό το κόστος, πρέπει να ληφθούν υπόψη οι παράγοντες που το διαμορφώνουν - αναφέρονται σε προηγούμενο κεφάλαιο.

Για παράδειγμα, όσον αφορά τον πρώτο παράγοντα, το κόστος μισθοδοσίας του προσωπικού δηλαδή, μια εταιρία μπορεί να χαρακτηρίζεται από υψηλό μισθολογικό κόστος λόγω νοοτροπίας ή φορολογίας και ασφαλιστικού, ενώ μια εταιρία, για τους ίδιους λόγους - ενδεχομένως σε διαφορετική χώρα ή ήπειρο - να έχει υποπολλαπλάσιο κόστος -αποδεικνύεται σε αρκετές εταιρίες του δυτικού κόσμου, οι οποίες προβαίνουν σε εξωτερικές αναθέσεις (outsourcing) ή ανοίγουν παραρτήματα σε χώρες με χαμηλό μισθολογικό κόστος, όπως η Ινδία ή η Κίνα.

Ο δεύτερος παράγοντας, τα πάγια έξοδα των κτιρίων, μπορεί να μειωθεί με μετεγκατάσταση της εταιρίας είτε σε κατάλληλη οικοδομική ζώνη που παρέχει κτιριακή επιφάνεια, νερό, ρεύμα και διαδίκτυο με μικρότερο κόστος, ή σε μετεγκατάσταση σε άλλη πόλη ή χώρα.

Ο τρίτος παράγοντας, το κόστος των υλικών για τη δημιουργία του πρωτοτύπου, αν και είναι συνήθως μικρότερο από τα δυο προηγούμενα, διαμορφώνει και αυτό το NRE, ιδιαίτερα αν απαιτούνται πολλά και μακροχρόνια πειράματα, δοκιμάζοντας πολλά διαφορετικά σενάρια ή σχεδιασμούς. Η μείωση του κόστους αυτού προϋποθέτει την οικονομικά συμφέρουσα προμήθεια υλικών, τα οποία ταυτόχρονα καλύπτουν τις απαιτήσεις ποιότητας ώστε να επιτευχθούν τα αναμενόμενα επίπεδα αξιοπιστίας. Ταυτόχρονα, για να επιτευχθεί η συντομότερη δυνατή παράδοση, πρέπει να ακολουθηθεί μια τεχνική που συνδυάζει συντόμευση του χρονικού διαστήματος, μείωση των άστοχων επιλογών κατά τη σχεδίαση και εξάλειψη των προβληματικών πρωτοτύπων. Αυτή η τεχνική ονομάζεται δεν είναι άλλη από τη μοντελοποίηση, η οποία αποτελεί πάντα το πρώτο βήμα σε οποιαδήποτε μεθοδολογία σχεδιασμού, είτε πρόκειται για Ε.Σ. είτε για οποιοδήποτε άλλο προϊόν.

2.2.4 Αρχές μοντελοποίησης

Αφού έγινε κατανοητή η σημασία της μοντελοποίησης, το επόμενο βήμα είναι η σωστή μέθοδος μοντελοποίησης. Όταν χρησιμοποιούνται αυθαίρετα μοντέλα, υπάρχει το πρόβλημα ότι τα αυτά είναι κατανοητά μόνο από τον δημιουργό τους και συνεπώς υπάρχει δυσκολία διαμοιρασμού σε άλλους και δυσλειτουργία στην επεξεργασία και το σχολιασμό από αυτούς. Η λύση σε αυτό είναι η δημιουργία ενός μοντέλου σε τυποποιημένη γλώσσα, κατανοητή από τους ενδιαφερόμενους, ώστε να χρησιμοποιηθεί για τις δικές τους ανάγκες. Επομένως, πρέπει η σχεδιαστική ομάδα να επιλέξει κάποια γλώσσα μοντελοποίησης και να τη χρησιμοποιεί έως την κατασκευή του προϊόντος ή ακόμη και μετά για βελτιστοποίηση και αναθεώρηση.

Υπάρχουν τέσσερις βασικές αρχές στη μοντελοποίηση:

1. Η επιλογή του μοντέλου επιδρά στον τρόπο που αντιμετωπίζεται το πρόβλημα και στη λύση που διαμορφώνεται.
2. Κάθε μοντέλο μπορεί να εκφραστεί με διαφορετικά επίπεδα ακρίβειας.
3. Τα πιο ρεαλιστικά μοντέλα είναι τα πιο επιτυχημένα.
4. Δεν υπάρχει μόνο ένα μοντέλο για κάθε περίπτωση και κρίνεται σημαντικό να χρησιμοποιούνται τα διαφορετικά μοντέλα προκειμένου να διαπιστώνονται πιθανές αποκλίσεις και να καθίστανται πιο αντικειμενικές οι προδιαγραφές των συστημάτων.

Η πρώτη αρχή υπαινίσσεται ότι η σχεδιαστική ομάδα θα πρέπει να γνωρίζει αρκετούς τρόπους μοντελοποίησης ώστε να επιλέξει το κατάλληλο εργαλείο. Μέσα από ένα όσο το δυνατό πιο ρεαλιστικό μοντέλο καθίσταται δυνατός ο εντοπισμός πτυχών του συστήματος που θα μπορούσαν να έχουν παραληφθεί από τους σχεδιαστές. Για παράδειγμα, μιας μορφής μοντελοποίηση ενός ηλεκτρικού οχήματος μπορεί να πραγματοποιηθεί με ορισμένες μετρήσεις σε έναν εργαστηριακό ηλεκτρικό κινητήρα, με τη βοήθεια ορισμένων οργάνων κατάλληλων για ηλεκτρικές μετρήσεις, όπως βολτόμετρα, αμπερόμετρα, βατόμετρα κ.ά. Ωστόσο, ο ενδεδειγμένος τρόπος μοντελοποίησης περιλαμβάνει την ανάπτυξη των μαθηματικών εξισώσεων που διέπουν ολόκληρο το σύστημα της ηλεκτρικής μηχανής και του οχήματος καθώς και την προσομοίωση του συστήματος αυτού σε κατάλληλο προσομοιωτή, με δυνατότητα επεξεργασίας-αλλαγής κάποιων παραμέτρων ώστε να διαπιστωθεί αν προσδίδει όφελος ή όχι η τροποποίηση. Εκείνος που θα επιλέξει το μοντέλο οφείλει να έχει γενική γνώση και να μην εξειδικεύεται σε ένα θέμα μόνο, όπως, για παράδειγμα, ένας σχε-

διαστής βάσης δεδομένων που μοντελοποιεί ένα σύστημα μπορεί να γνωρίζει μόνο μοντέλα οντοτήτων και εξαρτήσεων, ενώ ένας προγραμματιστής αντικειμενοστρεφών γλωσσών μπορεί να γνωρίζει μόνο για μοντέλα αντικειμένων και ιδιοτήτων.

Η δεύτερη αρχή επισημαίνει ότι το ίδιο στοιχείο μπορεί να εκφραστεί με πολλά επίπεδα λεπτομέρειας. Από ένα απλό εκτελέσιμο σύστημα που δέχεται κάποιες εισόδους και παράγει στην έξοδο κάποια αποτελέσματα (υψηλό επίπεδο), έως το κατώτατο επίπεδο ενός τρανζίστορ, όπου ανιχνεύεται η συμπεριφορά της πύλης όπου δέχεται τον παλμό λειτουργίας (gate). Συνήθως, ακολουθείται μια σχεδιαστική προσέγγιση που χαρακτηρίζεται ως «top to bottom & bottom to top», η οποία υλοποιείται ως εξής: όταν απαιτείται ο σχεδιασμός ενός προϊόντος, αρχικά αναγνωρίζεται η υψηλού επιπέδου περιγραφή και έπειτα ο σχεδιαστής κατευθύνεται προς το επίπεδο της υλοποίησης (top to bottom) για να εξαχθούν τα βασικά δομικά στοιχεία από τα οποία αυτή αποτελείται. Σε συνέχεια της εξαγωγής, ακολουθείται μια πορεία bottom to top, δηλαδή μοντελοποιείται κάθε δομικό στοιχείο ξεχωριστά, τα δομικά στοιχεία που έχουν μοντελοποιηθεί συνδέονται και μοντελοποιούν μεγαλύτερα, μέχρι η διαδικασία να οδηγηθεί στο ανώτερο επίπεδο που είναι το τελικό προϊόν.

Σύμφωνα με την τρίτη αρχή, το εκάστοτε μοντέλο που επιλέγεται πρέπει να προσεγγίζει σε πολύ μεγάλο βαθμό το πραγματικό σύστημα (διευκρινίζεται ωστόσο ότι όλα τα μοντέλα είναι σε κάποιον βαθμό προσεγγιστικά), ώστε να αποφεύγονται οι εσφαλμένες εκτιμήσεις των λειτουργικών σημείων. Όταν τα μοντέλα αποκλίνουν σημαντικά από την πραγματικότητα, τότε δεν είναι δυνατή η εξέταση μεγάλου εύρους συνθηκών λειτουργίας, και η μοντελοποίηση

περιορίζεται σε βασικό επίπεδο, το οποίο όμως δεν επιτρέπει τη διαμόρφωση των ορίων ανοχής. Παρότι σε όλα τα μοντέλα επιχειρείται απλοποίηση της πραγματικότητας, είναι σημαντικό να τηρούνται ορισμένα όρια στην απλούστευση αυτή (π.χ. γραμμικοποίηση των μη γραμμικών εξισώσεων ενός συστήματος), διότι ελλοχεύει ο κίνδυνος να επηρεαστεί αρνητικά η δημιουργία του προϊόντος.

Η τέταρτη αρχή διατείνεται ότι για το ίδιο προϊόν απαιτούνται πολλά μοντέλα που αποτυπώνουν μια διαφορετική εικόνα. Για παράδειγμα, για ένα κατασκευαστικό έργο, έστω μια οικοδομή, διατίθενται διαφορετικά μοντέλα που αφορούν τμήματά του, όπως η ηλεκτρολογική εγκατάσταση, η μελέτη θερμομόνωσης, η στατική μελέτη κ.ά.. Ομοίως, για έναν σχεδιασμό Ε.Σ. χρησιμοποιούνται διαφορετικά μοντέλα που διαφέρουν ως προς το κομμάτι του συστήματος που εξετάζουν (π.χ. υλικό, λογισμικό), αλλά συνδέονται με το σύνολο του συστήματος. Τυπικά μοντέλα που συναντώνται στα Ε.Σ. είναι τα ακόλουθα:

- μοντέλο των καταστάσεων χρήσης (use cases), που περιγράφει τις λειτουργικές προδιαγραφές του συστήματος,
- μοντέλο σχεδιασμού, που παρουσιάζει τα φυσικά και ηλεκτρονικά δομικά στοιχεία του κυκλώματος και τον τρόπο που συνδέονται,
- μοντέλο υλοποίησης, το οποίο αποτυπώνει τη μορφή του τελικού προϊόντος,
- μοντέλο λειτουργικής εμφάνισης, το οποίο παρέχει την πληροφορία της επικοινωνίας με τον τελικό χρήστη

- μοντέλα περιγραφής της λειτουργίας του λογισμικού (πρόγραμμα), σχεδιασμού και προσομοίωσης/εξομοίωσης.

2.3 Περιβάλλον μοντελοποίησης

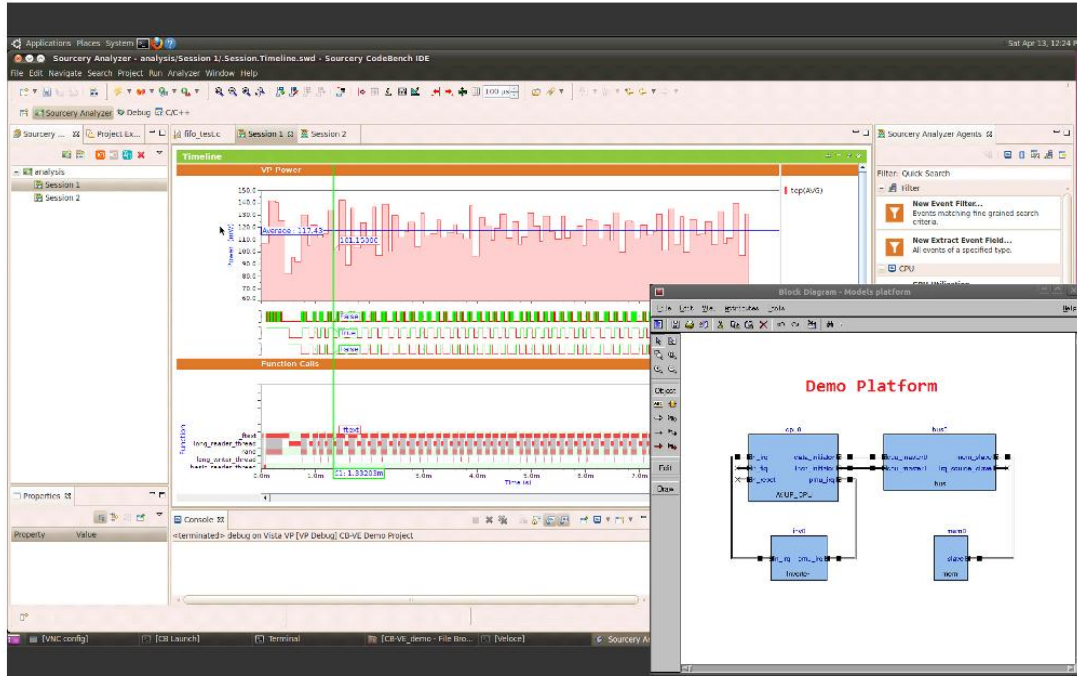
Τα εργαλεία μοντελοποίησης ενός Ε.Σ. μπορεί να είναι είτε μια τυπική γλώσσα προγραμματισμού, όπως η VHDL ή η Verilog, είτε μια σχηματική μορφή που μοιάζει με διάγραμμα, όπως η UML, είτε πολύπλοκα περιβάλλοντα που μπορούν να δεχθούν είτε γραφική είσοδο (block diagrams) και είσοδο σε μορφή κώδικα προγράμματος, όπως το Matlab.

Για τον σχεδιασμό Ε.Σ. διατίθενται πολλές γλώσσες ή εργαλεία μοντελοποίησης, με πιο διαδεδομένα τα παρακάτω:

- VHDL (VHSIC hardware description language ή γλώσσα περιγραφής υλικού VHSIC) [8],
- Verilog (hardware description language κατά IEEE 1364) [9],
- SystemC (hardware description language C++) [10],
- SystemVerilog (επέκταση της Verilog) [11],
- UML (Unified Modeling Language. Η μη δόκιμη απόδοση στην Ελληνική γλώσσα είναι Ενοποιημένη Γλώσσα Μοντελοποίησης. Αποτελεί την πλέον πρότυπη γλώσσα μοντελοποίησης στη μηχανική λογισμικού) [6],
- MATLAB (Matrix Laboratory) [12],
- EDA TOOLS (Electronic design automation) [13].

Η τάση της εποχής είναι η ανάπτυξη ειδικού περιβάλλοντος μοντελοποίησης που παρέχει τη δυνατότητα δημιουργίας ενός αφαιρετικού λειτουργικού μοντέλου πριν ακόμη σχεδιαστεί το σύστημα, το οποίο είναι χρήσιμο στον σχεδιασμό σύνθετων Ε.Σ. με διάφορους επεξεργαστές και δομικά στοιχεία από πολλούς πάροχους, οπότε πριν αποφασίσουν και επιλέξουν αυτά τα στοιχεία, επιβεβαιώνουν τη συνεργασία τους. Τα αντικείμενα (οντότητες) περιγράφονται σε οποιαδήποτε από τις προαναφερθείσες γλώσσες και καθορίζονται τα πρότυπα επικοινωνίας. Το εργαλείο χρησιμοποιεί συνήθως μοντέλο επιπέδου συναλλαγής (TLM, Transaction Level Modeling), το οποίο αποτελεί μια υψηλού επιπέδου μοντελοποίηση, όπου καθορίζονται τα δεδομένα που μεταφέρονται μεταξύ των διαφορετικών οντοτήτων του συστήματος και το οποίο περιλαμβάνει βιβλιοθήκες που παρέχουν διάφορους τύπους από επεξεργαστές, μνήμες, μικροελεγκτές, διαύλους επικοινωνίας και άλλα δομικά στοιχεία. Τα στοιχεία αυτά, αφού επιλεγούν από τις βιβλιοθήκες του περιβάλλοντος, διασυνδέονται κατάλληλα, ανάλογα με την απαίτηση της εφαρμογής, κι έτσι δημιουργείται το εκτελέσιμο αρχείο, το οποίο είναι ένας κώδικας στοιχείων (blocks) και εντολών που προσομοιώνουν το σύστημα. Η συχνότητα δειγματοληψίας που επιλέγεται για τη προσομοίωση επηρεάζει και την ευαισθησία του μοντέλου (όσο μεγαλύτερη συχνότητα τόσο μεγαλύτερη ευαισθησία του μοντέλου, άρα και ακρίβεια των αποτελεσμάτων). Τα αποτελέσματα του εκτελέσιμου αρχείου αποτυπώνονται με διαγράμματα που παρέχονται από το περιβάλλον. Σε αυτά περιλαμβάνονται οι πληροφορίες για τις εξόδους του συστήματος, όπως έξοδοι των συναρτήσεων για τις εισόδους που χρησιμοποιούνται, χρόνοι μεταφοράς των δεδομένων, εκτέλεση προτεραιότητας των διακοπών, χρόνοι εξυπηρέτησης διακοπών, σφάλματα/καθυστερήσεις κλήσης συναρτήσεων κ.ά. το πλεονέκτημα των αρχείων σε αυτού του είδους το

περιβάλλον προσομοίωσης, που είναι ουσιαστικά εκτελέσιμοι κώδικες, είναι ότι η τροποποίηση των παραμέτρων μπορεί να πραγματοποιηθεί άμεσα, με παρεμβάσεις στο ίδιο το εκτελέσιμο αρχείο. Πρόκειται για διαδικασία αποσφαλμάτωσης με πρόσβαση μέσα στο εσωτερικό των μοντέλων, με αποσφαλματωτή GNU (GNU Debugger) μέσω διεπαφής συμβατής με το GDB, για διόρθωση τμημάτων κώδικα. Ορισμένες πληροφορίες για την εκτέλεση του αρχείου μπορούν να εξαχθούν μέσω σχετικών συνοπτικών γραφημάτων που αφορούν τον ρυθμό ολοκλήρωσης αποτελεσμάτων (throughput), την καθυστέρηση επικοινωνίας (latency), την κατανάλωση ενέργειας (δυναμική, στατική, ρολογιού), την κατανομή ενέργειας (power distribution), τη χρήση μέγιστου, ελάχιστου και μέσου όρου του εύρους ζώνης διαύλων κ.ά. Έτσι, ο κώδικας που έχει δημιουργηθεί, ελεγχθεί, διορθωθεί από σφάλματα και τροποποιηθεί κατάλληλα μπορεί να ενσωματωθεί στο υλικό. Σε αυτό το στάδιο, από πλευράς λογισμικού, η διαδικασία της μοντελοποίησης έχει ολοκληρωθεί επιτυχώς (έχει επιτελέσει τον σκοπό της). Ασφαλώς, πιθανότατα να εμφανιστούν νέα σφάλματα τα οποία συνδέονται με το υλικό, αλλά τουλάχιστον θα έχει μειωθεί σημαντικά ο χρόνος έρευνας και ανάπτυξης, αφού θα μπορεί να συνδέεται το μοντέλο με το αντίστοιχο υλικό μέσω απομακρυσμένης αποσφαλμάτωσης (remote debugging) για να βρεθούν πιθανές αστοχίες. Η παραπάνω λειτουργία επιτυγχάνεται μέσα από το περιβάλλον εικονικής προτυποποίησης «Vista Virtual Prototyping», της εταιρίας Mentor.



Σχήμα 2-3 Μοντελοποίηση στην πλατφόρμα Vista Virtual Prototyping, της Mentor [1]

ΚΕΦΑΛΑΙΟ 3: ΟΡΓΑΝΩΣΗ ΚΑΙ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΩΝ Ε.Σ.

3.1 Εισαγωγή

Στο κεφάλαιο αυτό αναλύονται ζητήματα οργάνωσης και αρχιτεκτονικής που διέπουν τα Ε.Σ, τα οποία ασφαλώς βασίζονται στις ίδιες αρχές που διέπουν κάθε προγραμματιζόμενο σύστημα. Διευκρινίζεται στο σημείο αυτό ότι η αρχιτεκτονική αναφέρεται σε εκείνες τις ιδιότητες που είναι ορατές στον προγραμματιστή, για παράδειγμα το σετ εντολών, ο αριθμός των bit που χρησιμοποιούνται για την αναπαράσταση των δεδομένων, οι μηχανισμοί εισόδου/εξόδου, οι τεχνικές διευθυνσιοδότησης, κ.ά. Η οργάνωση απ' την άλλη είναι ο τρόπος με τον οποίο τα χαρακτηριστικά υλοποιούνται (λειτουργικές μονάδες και οι διασυνδέσεις τους), όπως τα σήματα ελέγχου, οι διεπαφές, η τεχνολογία των μνημών, κ.ά. [5]

Ακολούθως, συνοψίζονται οι βασικές αρχές οργάνωσης και αρχιτεκτονικής, που βρίσκουν εφαρμογή στη συγκεκριμένη κατηγορία συστημάτων, περιγράφοντας βασικές γνώσεις για τα 3 δομικά τμήματά τους, που είναι η μνήμη, ο επεξεργαστής και οι είσοδοι/έξοδοι. Τα δεδομένα αποθηκεύονται ή ανακαλούνται από τη μνήμη, επεξεργάζονται και μετασχηματίζονται από τον επεξεργαστή και μεταφέρονται μέσω διαύλων και δικτύων διασύνδεσης για τις εισόδους ή τις εξόδους.

3.2 Η μνήμη

Όπως οι μικροελεγκτές, έτσι και τα Ε.Σ. περιλαμβάνουν μια ποικιλία τεχνολογιών και ιδιοτήτων μνήμης, με την καθεμία να επιτελεί συγκεκριμένη λειτουργία, όπως η μνήμη SRAM για τους καταχωρητές, η μνήμη RAM για την εκτέλεση του

προγράμματος, η μνήμη FLASH για την αποθήκευση του προγράμματος στο ολοκληρωμένο και η μνήμη EEPROM για την αποθήκευση ρυθμίσεων. Στη συνέχεια του κεφαλαίου περιγράφονται όλες αυτές οι τεχνολογίες μνημών που απαντώνται σε όλα τα Ε.Σ.

Η μνήμη σε ένα Ε.Σ. χρησιμοποιείται είτε για τη διατήρηση των δεδομένων για όσο διατηρείται η τροφοδοσία (πτητική μνήμη) είτε μόνιμα (μη πτητική μνήμη) για τις εντολές προγράμματος του επεξεργαστή. Συχνά χρησιμοποιείται ένα μίγμα τεχνολογιών μνήμης μέσα σε ένα Ε.Σ., ώστε να υπάρχει ευελιξία για την εκάστοτε εφαρμογή. Ένα είδος μνήμης διατηρεί το περιεχόμενό της ακόμα κι όταν δεν τροφοδοτείται και γι' αυτό ονομάζεται μη πτητική (non volatile), ενώ η αντίθετη περίπτωση, δηλαδή η μνήμη που δε διατηρεί το περιεχόμενό της όταν πάψει να τροφοδοτείται ονομάζεται πτητική (volatile).

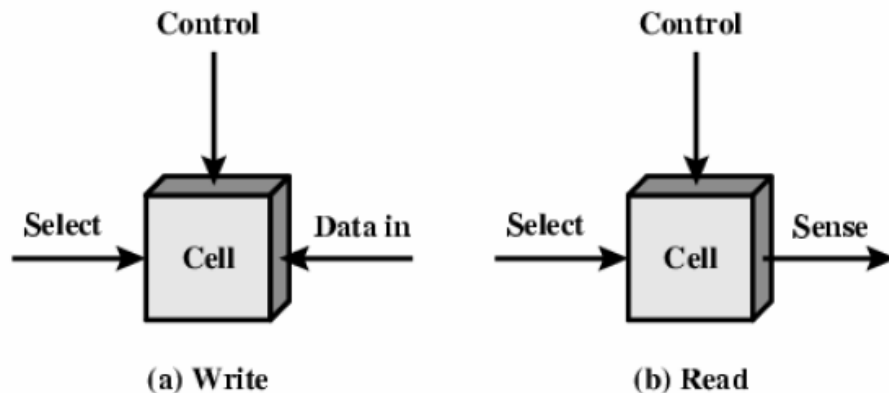
Ορισμένες συσκευές μνήμης έχουν μεγάλη χωρητικότητα, όμως απαιτούν στοιχεία κυκλώματος πρόσθετης υποστήριξης για την ανανέωση ή τον έλεγχο και γι' αυτό η πρόσβαση σε αυτές είναι αργή. Αντίθετα, άλλες συσκευές μνήμης υστερούν σε χωρητικότητα προκειμένου να επιτυγχάνουν μεγαλύτερη ταχύτητα προσπέλασης, και να είναι σε θέση να ανταπεξέλθουν στις απαιτήσεις ενός γρηγορότερου επεξεργαστή.

Οποιαδήποτε μνήμη αποτελείται από κελιά (cells) και το κάθε κελί αποθηκεύει ένα bit. Παρουσία τάσης οδηγεί στη αποθήκευση του λογικού bit 1 στο κατάλληλο κελί της μνήμης, ενώ απουσία τάσης οδηγεί στην αποθήκευση του λογικού bit 0. Στην περίπτωση πιο σύνθετων υλοποιήσεων, χρησιμοποιούνται στο ίδιο κελί διαφορετικά επίπεδα τάσης (MLC, Multi-Level Cell ή κελί πολλαπλών επιπέδων τάσης) και κατ' αυτόν τον τρόπο επιτρέπουν την αποθήκευση 2 ή περισσότερων

bit σε ένα κελί του ενός bit ,αυξάνοντας έτσι ουσιαστικά τη χωρητικότητα του κελιού (επιφάνεια πυριτίου). Οι τελευταίες βρίσκονται σε ερευνητικό επίπεδο, διότι απαιτούνται ανιχνευτές τάσης μεγάλης ευαισθησίας και είναι επιρρεπείς σε τροποποίηση και αλλοίωση των αποτελεσμάτων, ιδιαίτερα για τις πολύ μικρές τεχνολογίες υλοποίησης.

Τα bit ομαδοποιούνται συνήθως σε λέξεις των 8 στοιχείων (1 Byte) ή και σε μεγαλύτερες όπως των 32 Byte. Συνεπώς, μνήμη m λέξεων, με λέξεις των n bit η κάθε μια, φέρει συνολικά $m \cdot n$ bit. Η αναφορά σε μια μνήμη γίνεται ως μνήμη m x n (ή m από n). Ορισμένες μνήμες διαθέτουν bit ελέγχου, για την αποφυγή σφαλμάτων κατά την αποθήκευση των δεδομένων και των εντολών, οι οποίες χρησιμοποιούν κώδικα ECC (Error Correcting Code, κώδικα για διόρθωση σφαλμάτων) και χρησιμοποιούνται σε περιπτώσεις σύνθετων λειτουργικών καταστάσεων. [5]

Μια μνήμη αποτελείται από m λέξεις των n bit. Η διευθυνσιοδότηση αυτών των m λέξεων είναι απαραίτητη προκειμένου να διαβαστεί η λέξη στην έξοδο δεδομένων της μνήμης. Τα σήματα εισόδου διευθύνσεων $\log_2(m)$ είναι απαραίτητα για τον προσδιορισμό μιας συγκεκριμένης λέξης. Έτσι, εάν μια μνήμη έχει k διευθύνσεις ως είσοδο, αυτή μπορεί να αποθηκεύσει μέχρι 2^k λέξεις. Ο διάυλος των διευθύνσεων απαιτεί k bit. Αυτά μπορεί να βρίσκονται σε k διαφορετικές γραμμές ή σε λιγότερες με χρήση πολυπλέκτη. Η διευθυνσιοδότηση μιας μνήμης επιτρέπει την εγγραφή ή ανάγνωση μιας λέξης.



Σχήμα 3-1 Η λειτουργία των κυττάρων μνήμης [5]

Μερικές μνήμες μπορούν μόνο να προσπελαστούν (όπως η ROM, Read Only Memory), ενώ άλλες επιτρέπουν την προσπέλαση και την εγγραφή δεδομένων (όπως η RAM, Random Access Memory). Σε αντίθεση με τη μνήμη μόνο για ανάγνωση, δεν υπάρχει μνήμη μόνο για εγγραφή (δηλαδή να μην επιτρέπει να γίνει ανάγνωση).

Κοινό στοιχείο για όλες τις μνήμες αποτελεί ότι μπορούν να διευθυνσιοδοτούν λέξεις. Ένα στοιχείο που τις διαφοροποιεί είναι το πόσα bit ή Byte έχει κάθε λέξη. Συνήθως μια λέξη θεωρείται μια ακολουθία των 8 bit ή 32 bit, γεγονός που υποδεικνύει ότι με μια εντολή διεύθυνσης μνήμης ο επεξεργαστής αποκτά πρόσβαση σε 8 bit ή 32 bit αντίστοιχα. Συνεπώς, απαιτούνται 4 διαδοχικές διευθύνσεις για τη μεταφορά 32 των bit, όταν χρησιμοποιούνται τα 8 bit. Υπάρχουν όμως και ειδικοί τρόποι πρόσβασης σε κάποιες μνήμες, κατά τους οποίους στέλνεται μόνο η αρχική διεύθυνση και το συνολικό μέγεθος των Byte που απαιτούνται για την ανάγνωση (μεταφορά ριπής, burst transfer).

Τα ολοκληρωμένα τσιπ μνήμης κατασκευάζονται σε διαφορετικά μεγέθη οργάνωσης. Παραδείγματος χάριν, ένα τσιπ DRAM (δυναμική RAM) μπορεί να περιγράψει ως 4 MB x 1 (οργανωμένο σε bit), ενώ ένα τσιπ SRAM (στατική RAM) μπορεί να είναι 512 KB x 8 (οργανωμένο σε λέξεις). Η ικανότητα αποθήκευσης και στις δύο περιπτώσεις είναι η ίδια. Αυτό το οποίο διαφέρει είναι η οργάνωση του κάθε τσιπ μνήμης. Στην περίπτωση της DRAM, απαιτούνταν οκτώ τσιπ συνδεδεμένα όλα στις ίδιες γραμμές διευθύνσεων, όπου το πρώτο τσιπ δίδει το bit ελάχιστης σηματικότητας (LSB, Less Significant Bit), ενώ το τελευταίο τσιπ μεταφέρει το μεγαλύτερης σηματικότητας bit (MSB, Most Significant Bit). Η πρόσβαση σε αυτά τα τσιπ είναι παράλληλη και μόλις αποκωδικοποιηθεί η διεύθυνση μνήμης, και τα 8 bit τοποθετούνται ταυτόχρονα στην έξοδο.

Στην περίπτωση της SRAM απαιτείται μόνο ένα τσιπ. Εντούτοις, επειδή συνήθως τα τσιπ DRAM οργανώνονται παράλληλα, υποστηρίζουν ταυτόχρονη προσπέλαση και έτσι επιτυγχάνονται υψηλές ταχύτητες πρόσβασης. Το μέγεθος της DRAM είναι (4MBx1)x 8 ή 32 MB.

3.2.1 Μνήμη τυχαίας προσπέλασης

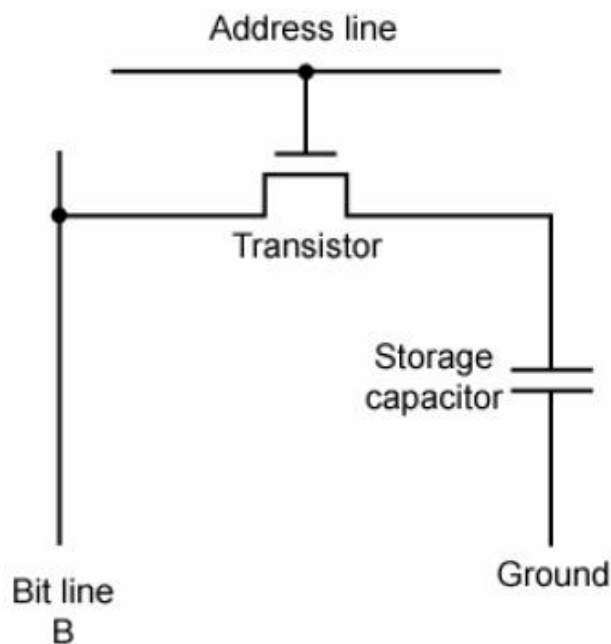
Σε αυτό το είδος μνήμης, ο χρόνος μεταξύ αίτησης και λήψης των δεδομένων θεωρείται από τη στιγμή που τροφοδοτείται μία διεύθυνση στη μνήμη μέχρι τη στιγμή που τα δεδομένα εκείνα θα έχουν αποθηκευτεί ή θα είναι διαθέσιμα προς χρήση και είναι προβλέψιμος. Η μνήμη RAM αναπτύχθηκε με τους πρώτους υπολογιστές με τρανζίστορ της δεκαετίας του 1960, ως συνέχεια της μνήμης μαγνητικού πυρήνα (magnetic core memory) της εποχής, της οποίας η ανάγνωση και η εγγραφή ήταν αρκετά αργή. Επίσης, οι μαγνητικές ταινίες της εποχής

υποστήριζαν μόνο τη σειριακή εγγραφή και ανάγνωση. Έτσι, η «γρήγορη» μνήμη RAM, έδωσε μια σημαντική ώθηση στους στις δυνατότητες των σύγχρονων υπολογιστικών συστημάτων.

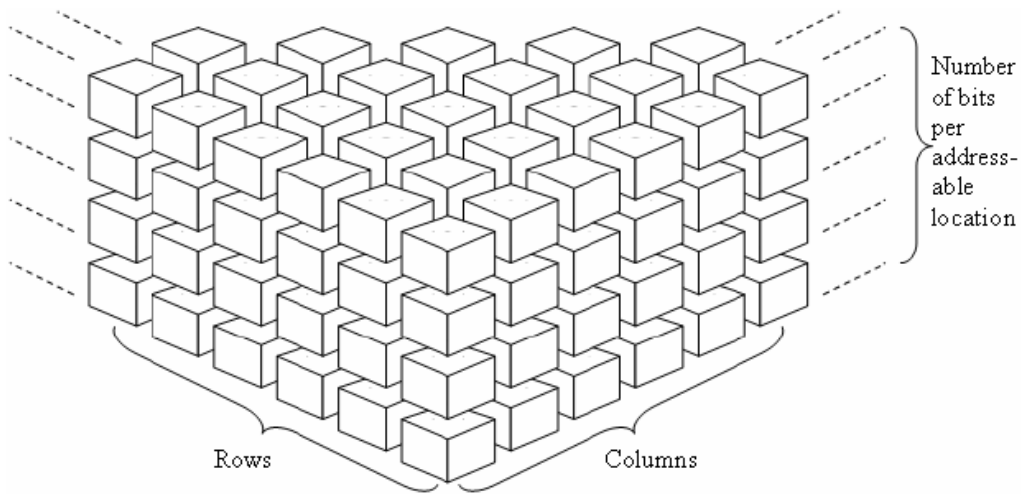
Η RAM, ως προσωρινή μνήμη λειτουργίας στους ηλεκτρονικούς υπολογιστές(λειτουργεί όσο τροφοδοτείται ο υπολογιστής με ενέργεια), είναι το μέρος όπου ο επεξεργαστής των Ε.Σ. εγγράφει τα δεδομένα για προσωρινή αποθήκευση. Η RAM είναι πτητική, δηλαδή χάνει το περιεχόμενό της όταν πάψει να τροφοδοτείται το σύστημα. Οποιοσδήποτε πληροφορίες αποθηκεύονται στη RAM και πρέπει να διατηρηθούν, γράφονται σε κάποια μνήμη μόνιμης αποθήκευσης, όπως ένας σκληρός δίσκος. Προκειμένου να αντιμετωπιστεί το μειονέκτημα της πτητικότητας των δεδομένων, που είναι κρίσιμης σημασίας σε διακομιστές, έχουν κατασκευαστεί μνήμες RAM με εφεδρικό σύστημα τροφοδότησης ενέργειας, ώστε ακόμα και αν διακοπεί η τροφοδοσία, εκείνες να συνεχίσουν να τροφοδοτούνται ώστε να διατηρούν τα δεδομένα τους.

Οι μνήμες RAM διαιρούνται σε δύο οικογένειες: τη στατική RAM (γνωστή ως Static RAM ή SRAM) και τη δυναμική RAM (γνωστή ως dynamic RAM ή DRAM). Οι μνήμες SRAM χρησιμοποιούν λογικές πύλες συνδεσμολογίας μανδαλωτών (latches) ή flip-flop. Αυτή η κατηγορία RAM επιτυγχάνει τις μεγαλύτερες ταχύτητες, απαιτεί ελάχιστα εξωτερικά στοιχεία κυκλώματος υποστήριξης και έχει σχετικά μικρή κατανάλωση ενέργειας. Μειονεκτεί ωστόσο από πλευράς χωρητικότητας (απαιτείται εξαπλάσια επιφάνεια σε σύγκριση με το κύτταρο της DRAM), άρα και πυκνότητας ολοκλήρωσης, καθώς και κόστους (υψηλότερο συγκριτικά με τη DRAM).Γι' αυτό, στην πράξη αυτή η μνήμη χρησιμοποιείται σε συγκεκριμένου σκοπού κυκλώματα στα οποία η μεγάλη ταχύτητα πρόσβασης στη μνήμη είναι ζωτικής σημασίας (όπως συμβαίνει με τις

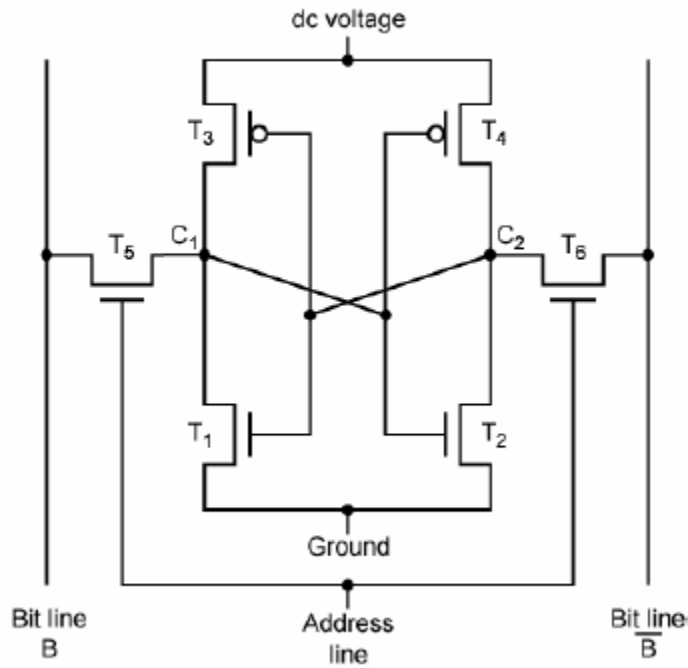
κρυφές μνήμες). Η DRAM χρησιμοποιεί πυκνωτές για την αποθήκευση φορτίων αντί για τρανζίστορ, και αισθητήρια για την διαπίστωση της ύπαρξης ή όχι φορτίου. Οι συστοιχίες των πυκνωτών διατηρούν το φορτίο τους μόνο για μια μικρή χρονική περίοδο προτού αποφορτιστούν. Επομένως, η DRAM απαιτεί συνεχή ανανέωση (refresh), δηλαδή το κελί διαβάζεται και ξαναγράφεται κάθε λίγα χιλιοστά του δευτερολέπτου. Η απαίτηση διαρκούς ανανέωσης αποτελεί παράγοντα εισαγωγής επιπλέον καθυστέρηση της πρόσβασης στη μνήμη, αφού συχνά εμφανίζεται η περίπτωση αναμονής μιας ανάγνωσης που απαιτείται εάν την ίδια στιγμή πραγματοποιείται αναπόφευκτα ανανέωση του κελιού. Οι DRAM είναι συσκευές μνήμης υψηλής χωρητικότητας, διαθέσιμες σε μια ευρεία ποικιλία υποκατηγοριών. Λόγω της πολυπλοκότητας που έχουν μπορούν να διασυνδεθούν μόνο με επεξεργαστές, και δεν είναι πρακτική η σύνδεση με μικροελεγκτές σε Ε.Σ. Στους επεξεργαστές όπου διευθυνσιοδοτείται σημαντικό μέγεθος μνήμης, υπάρχει εγγενώς υποστήριξη DRAM (ή υπάρχει η δυνατότητα σύνδεσης σε ενδιάμεσο ελεγκτή μνήμης).



Σχήμα 3-2 Κύτταρο δυναμικής μνήμης [5]



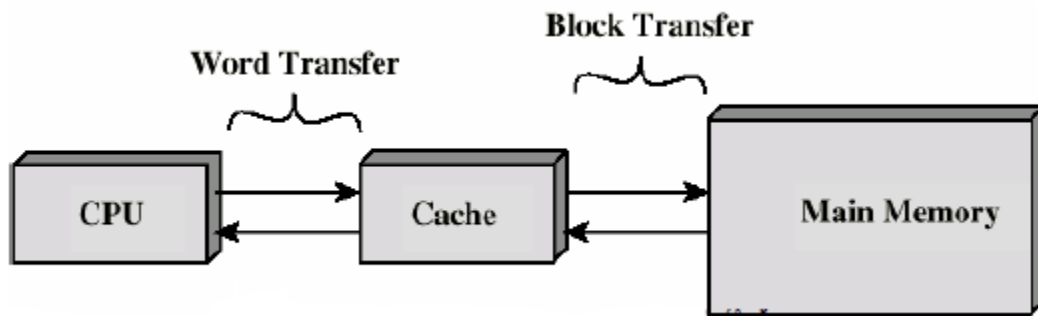
Σχήμα 3-3 Οργάνωση της δυναμικής μνήμης [5]



Σχήμα 3-4 Κύτταρο στατικής μνήμης [5]

Ένα σημαντικό πρόβλημα στη χρήση της DRAM προκύπτει από το γεγονός ότι είναι εξωτερική μνήμη (δε βρίσκεται στο ίδιο ολοκληρωμένο με τον επεξεργαστή),

συνεπώς υπάρχει αυξημένη καθυστέρηση και ενεργειακές απαιτήσεις σε κάθε πρόσβαση. Για την άρση αυτού του μειονεκτήματος, κάποιοι επεξεργαστές έχουν κρυφές μνήμες (cache memories) εντολών και δεδομένων πάνω στο ίδιο ολοκληρωμένο με τον επεξεργαστή και αποθηκεύουν τις πρόσφατες προσβάσεις μνήμης. Αυτές οι μνήμες είναι συχνά εσωτερικές των επεξεργαστών και υλοποιούνται με γρήγορα στοιχεία μνήμης και μεγάλης ταχύτητας διαύλους μεταφοράς δεδομένων. Έτσι, τα δεδομένα που επαναχρησιμοποιούνται μπορούν να αποθηκευτούν προσωρινά σε αυτήν τη μνήμη και την επόμενη φορά που θα ζητηθούν να βρίσκονται κοντά στον επεξεργαστή. Οι κρυφές μνήμες είναι τεχνολογίας SRAM.



Σχήμα 3-5 Κρυφή μνήμη-Cache memory [5]

Όσον αφορά τις μνήμες DRAM, μια αύξηση των επιδόσεων επήλθε όταν η μνήμη συγχρονίστηκε με το ρολόι του διαύλου που συνδέει τον επεξεργαστή με τη μνήμη, προκειμένου η συνολική επικοινωνία να είναι συγχρονισμένη. Αυτή η μνήμη ονομάζεται synchronous DRAM (SDRAM), μεταχύτητα λειτουργίας στα MHz.

Μια βελτίωση στη SDRAM πραγματοποιήθηκε όταν άλλαξε η επικοινωνία με τη

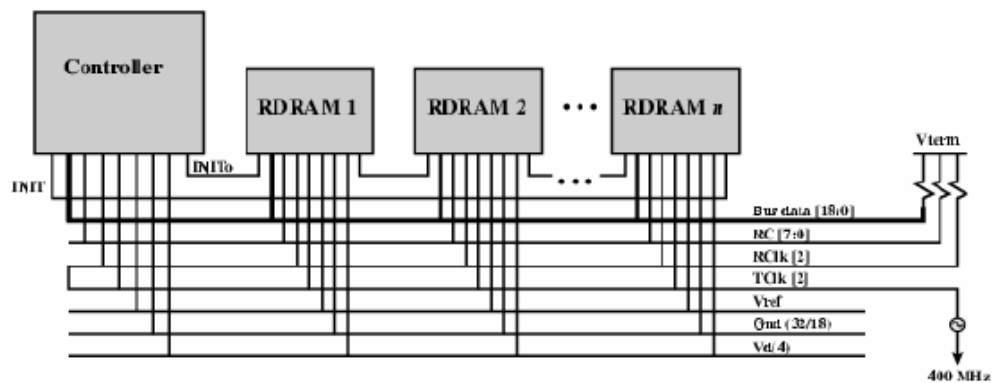
μνήμη, και από εκεί που γίνονταν με τις λογικές στάθμες (1 και 0), στη συνέχεια γίνονταν με τις ανερχόμενες και κατερχόμενες ακμές του ρολογιού (rising, falling edge). Έτσι, ο ρυθμός επικοινωνίας με τη μνήμη διπλασιάστηκε (DDR , Double Data Rate,), παρότι το ρολόι του συστήματος λειτουργεί στην ίδια συχνότητα. Περαιτέρω βελτίωση συντελέστηκε με την αύξηση της συχνότητας λειτουργίας του ρολογιού, μείωσης της τάσης τροφοδοσίας (άρα και των απωλειών του συστήματος) και τον διπλασιασμό της χωρητικότητας του διαύλου δεδομένων (από 128 bit σε 256 bit). Η επόμενη γενιά των μνημών (DDR3 SDRAM) διπλασίασε το εύρος του διαύλου σε 512 bit, και την υποβίβασε την τάση λειτουργίας στο 1.5 το εύρος του διαύλου σε 512 bit, και την τάση λειτουργίας στα 1.5 Volt και συχνότητα λειτουργίας DDR3-1600. Η ακόλουθη γενιά, που ακόμη δεν έχει καθιερωθεί στην αγορά γιατί είναι σε πρώιμο στάδιο, είναι η DDR4, με 1.2 Volt και με συχνότητες λειτουργίας DDR4-2133 και DDR4-3200. Οι μνήμες της τελευταίας γενιάς έχουν τόσο μεγάλες ταχύτητες, οι οποίες πρακτικά μπορούν να καταστούν εκμεταλλεύσιμες μόνο από εφαρμογές που κάνουν εντατική χρήση της μνήμης (εφαρμογές διακομιστών), για αυτό και η αύξηση της απόδοσης σε ένα σύστημα που αναβαθμίζεται από DDR3 σε DDR4 κυμαίνεται από 0 έως 5%, ενώ το κόστος αγοράς της μνήμης είναι 2 με 3 φορές περισσότερο.

Επίσης, διατίθενται και τροποποιήσεις των DDR, όπως οι:

- LPDDR(X) (όπου X=2, 3, 4 και LP Low-Power, χαμηλή κατανάλωση), δηλαδή μνήμες που λειτουργούν με μειωμένη τάση τροφοδοσίας και χρησιμοποιούνται σε φορητές συσκευές (π.χ. σε κινητά τηλέφωνα, τάμπλετς κ.ά.),
- GDDR(X) (όπου X=2, 3, 4, 5 και G Graphics, γραφικά), οι οποίες

χρησιμοποιούνται σε κάρτες γραφικών και φέρουν κάποιες εξειδικευμένες αρχιτεκτονικές τροποποιήσεις,

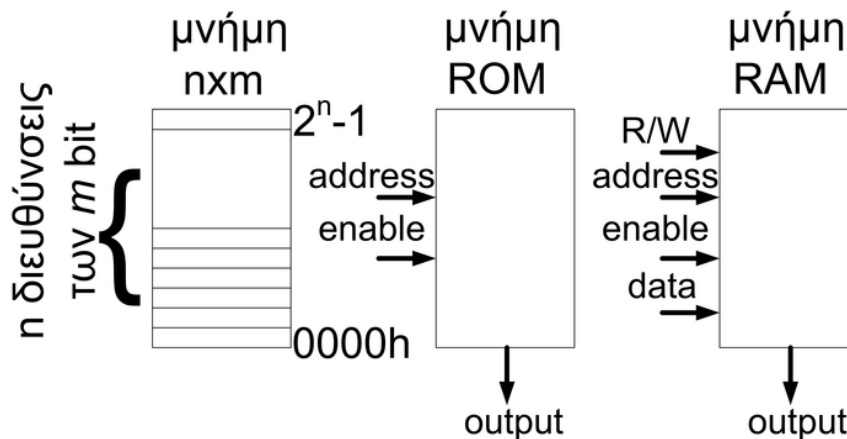
- Rambus RDRAM και η εξέλιξή της ως XDR SDRAM, οι οποίες είναι κλειστές τεχνολογίες με ευρεσιτεχνίες που κατέχει η Rambus και υποστηρίζουν μεγαλύτερες συχνότητες λειτουργίας από τις αντίστοιχες DDR, αλλά με αυξημένο κόστος,
- τεχνολογίας Υβριδικού Κύβου Μνήμης (Hybrid Memory Cube, HMC) που συναγωνίζεται την τεχνολογία Μνήμη Υψηλού Εύρους ζώνης (High Bandwidth Memory, HBM), αλλά βρίσκονται αμφότερες σε ερευνητικό επίπεδο (μετά τη DDR4). Οι τεχνολογίες αυτές χαρακτηρίζονται από την κάθετη τοποθέτηση πολλών ολοκληρωμένων, την κατάλληλη σύνδεσή τους με οπές (vias), τον μειωμένο αριθμό των γραμμών μεταφοράς δεδομένων και από υψηλής ταχύτητας σειριακή αποστολή και λήψη δεδομένων από τον επεξεργαστή.



Σχήμα 3-6 Διάγραμμα Rambus [5]

3.2.2 Μνήμη Ανάγνωσης μόνο

Η μνήμη μόνο για ανάγνωση (ROM, Read Only Memory) είναι μια μνήμη που μπορεί να προσπελαστεί για ανάγνωση, αλλά όχι να αποθηκεύσει δεδομένα κατά την τυπική λειτουργία. Φυσικά, υπάρχει τρόπος να γραφούν μια ή ελάχιστες φορές δεδομένα, αλλά αυτή η διαδικασία απαιτεί ειδικό προγραμματισμό και όχι τυπική αποθήκευση, η οποία γίνεται συνήθως από τον κατασκευαστή, προτού τοποθετηθεί στο κατάλληλο τμήμα του Ε.Σ. Στο Σχ.3-7 παρέχεται ένα διάγραμμα δομικών στοιχείων μιας ROM.



Σχήμα 3-7 Απεικόνιση διευθυνσιοδότησης μνήμης ROM και μνήμης RAM [1]

Σε μικρά Ε.Σ., μια μνήμη τύπου ROM συνήθως περιλαμβάνει ολόκληρο τον όγκο δεδομένων του προγράμματος λειτουργίας. Ωστόσο, για μεγαλύτερων δυνατοτήτων Ε.Σ., η ROM μπορεί να χρησιμοποιηθεί ποικιλοτρόπως, όπως για παράδειγμα, για το πρόγραμμα εκκίνησης (boot loader), δηλαδή το πρώτο πρόγραμμα που εκτελείται από ένα Ε.Σ. μόλις τροφοδοτηθεί και στη συνέχεια το πρόγραμμα αυτό να μεταφορτώσει το υπόλοιπο πρόγραμμα ή το λειτουργικό σύστημα από κάποιο μέσο μεγαλύτερης χωρητικότητας, όπως μια μνήμη Flash.

Μια ακόμη χρήση της ROM αφορά στην υλοποίηση ενός συνδυαστικού

κυκλώματος. Οποιαδήποτε συνδυαστική συνάρτηση k μεταβλητών μπορεί να υλοποιηθεί με τη χρησιμοποίηση μιας $2^k \times 1$ ROM, και μπορούν να υλοποιηθούν n λειτουργίες των ίδιων μεταβλητών k χρησιμοποιώντας μια $2^k \times n$ ROM.

Η διαδικασία της φόρτωσης του λογισμικού σε μία ROM είναι γνωστή με τον όρο «burn» (ο ελληνικός μη δόκιμος όρος είναι «κάψιμο»). Αν η ROM είναι ειδικού τύπου, μπορεί να προγραμματιστεί εντός συστήματος (ISP, In-System Programming) ή εντός κυκλώματος (ICP, In-Circuit Programming). Η απλή μνήμη ROM χρησιμοποιείται σε Ε.Σ. τα οποία δεν προορίζονται για κάποια ενημέρωση ή μεταβολή στο επίπεδο της ROM κατά τη διάρκεια της λειτουργίας τους, γιατί κατά τον προγραμματισμό τους το κύκλωμα της μνήμης διαμορφώνεται μόνιμα.

HPROM (Programmable ROM) είναι μια κατηγορία ROM η οποία μπορεί να προγραμματιστεί μια φορά από τον χρήστη (γρήγορη προτυποποίηση με συγκεκριμένη διαδικασία), γεγονός που τις καθιστά καταλληλότερες για Ε.Σ. περιορισμένων απαιτήσεων.

Η διαδικασία προγραμματισμού μια μνήμης PROM από τον χρήστη περιλαμβάνει τα εξής βήματα: 1) το επιθυμητό περιεχόμενο της μνήμης καθορίζεται από κατάλληλα διαμορφωμένο αρχείο, 2) το αρμόδιο για τον προγραμματισμό της ROM κομμάτι του εξοπλισμού εκτελεί τις απαιτούμενες συνδέσεις βάσει αρχείου, 3) οι αχρείαστες διασυνδέσεις καταστρέφονται με τη διέλευση υψηλού ρεύματος σε σχέση με το ρεύμα κανονικής λειτουργίας, 4) οι διασυνδέσεις που παραμένουν υλοποιούν το κύκλωμα. Οι συνδέσεις που καταστρέφονται δε μπορούν να επανεγκατασταθούν. Για αυτόν τον λόγο, η βασική PROM αναφέρεται συχνά ως μιας φορές χρονοπρογραμματίσιμη συσκευή (OTP, One Time Programmable).

3.2.3 Erasable Programmable Memory

Ένας άλλος τύπος του PROM είναι μια διαγράψιμη PROM (EPROM, Erasable PROM). Σε αυτήν το ιδιαίτερο χαρακτηριστικό είναι ότι η πύλη (gate) των τρανζίστορ που χρησιμοποιούνται δεν είναι σταθερή στο υλικό αλλά «αιωρούμενη» (floating). Κατά τον προγραμματισμό, τα τρανζίστορ συνδέονται μέσω ροής ηλεκτρονίων (με εφαρμογή υψηλότερου από το κανονικό επιπέδου τάσης παλμού στην πύλη, από 12V έως 25V), με το υπόλοιπο κύκλωμα. Με την εφαρμογή του απαιτούμενου επιπέδου τάσης καθίσταται δυνατός ο επαναπρογραμματισμός της.

Οι OTP ROM χρησιμοποιούνται όλο και λιγότερο στα Ε.Σ., αφού δεν είναι ευέλικτες. Αν απαιτηθεί κάποια αλλαγή πρέπει να απομακρυνθεί και να πεταχτεί το παλιό ολοκληρωμένο και να προγραμματιστεί ένα νέο. Έτσι, η χρήση των OTP ROM είναι μια ακριβή επιλογή ανάπτυξης Ε.Σ., και προτιμώνται άλλα είδη μνήμης.

Η καλύτερη επιλογή για την ανάπτυξη και τη διόρθωση συστημάτων είναι η διαγράψιμη προγραμματιζόμενη μνήμη μόνο για ανάγνωση (EPROM). Το περιεχόμενο της μνήμης EPROM καθαρίζεται με την εφαρμογή στη μια πλευρά του ολοκληρωμένου ενός υπερϊόδους σήματος, αφήνοντας το πεδίο ελεύθερο για εκ νέου προγραμματισμό και χρησιμοποίηση. Η τεχνολογία της εν λόγω μνήμης συμβαδίζει με εκείνη των OTP ROM ως προς τον προγραμματισμό και την αποδοχή των σημάτων διαμόρφωσης. Ως εκ τούτου, μια EPROM που χρησιμοποιείται στα στάδια της ανάπτυξης μιας εφαρμογής, μπορεί στη συνέχεια

να αντικατασταθεί από μια OTP ROM, χωρίς επιπλέον τροποποιήσεις του υπόλοιπου συστήματος.

Η συνήθης χωρητικότητα των μνημών EPROM και OTP είναι της τάξης των μερικών KB και μπορεί να φτάσει έως ένα ή μερικά MB.

3.2.4 Electrically Erasable Programmable Memory

Παρότι η παρούσα μνήμη είναι τύπου μνήμης μόνο για ανάγνωση, μπορεί να διαγραφεί και να προγραμματιστεί από τον χρήστη. Σπανίως καλείται επίσης και ηλεκτρικά μεταβλητή μνήμη μιας ανάγνωσης (EAROM , Electrically alterable read-only memory).

Η EEPROM μπορεί να διαγραφεί και να επαναπρογραμματιστεί μέσα σε ένα κύκλωμα, σε αντίθεση με τις προηγούμενες μνήμες, στις οποίες απαιτείται η απομάκρυνση από το κύκλωμα και η τοποθέτηση σε ειδική συσκευή. Η χωρητικότητα τους είναι σημαντικά μικρότερη από την τυποποιημένη ROM (μόνο μερικά KB), και έτσι δε χρησιμοποιούνται για αποθήκευση κώδικα προγραμματισμού. Η χρήση τους έγκειται στην αποθήκευση των παραμέτρων των συστημάτων και των πληροφοριών που χρίζουν διατήρησης κατόπιν της αποσύνδεσης.

Είναι κοινό για πολλούς μικροελεγκτές να ενσωματώνουν ένα μικρό ολοκληρωμένο EEPROM για την αποθήκευση των παραμέτρων του συστήματος. Η τεχνική αυτή χρησιμεύει ιδιαίτερα στα Ε.Σ. όπου απαιτείται αποθήκευση των

δικτυακών διευθύνσεων, των διευθύνσεων των αρχείων οι οποίες περιλαμβάνουν αριθμούς με αύξουσα καταχώρηση κ.ά.

3.2.5 Η μνήμη Flash

Η Flash είναι η νεότερης τεχνολογίας ROM. Σε πολλές περιπτώσεις συνυπάρχει μαζί με τη μνήμη EEPROM, αλλά σε κάποια Ε.Σ. την έχει αντικαταστήσει τελείως. Η μνήμη Flash έχει τα πλεονεκτήματα της επαναπρογραμματισιμότητας εντός κυκλώματος, όπως η EEPROM, και της μεγάλης χωρητικότητας, όπως η ROM. Τα ολοκληρωμένα της Flash αναφέρονται μερικές φορές ως «Flash ROM» ή «Flash RAM». Η Flash οργανώνεται κανονικά όπως οι τομείς σε ένα μαγνητικό δίσκο, και έχει το πλεονέκτημα ότι οι μεμονωμένοι τομείς μπορούν να διαγραφούν και να επανεγγραφούν χωρίς να επηρεάζεται το περιεχόμενο της συσκευής.

Υπάρχουν αρκετές διαφορετικές τεχνολογίες Flash, και οι απαιτήσεις διαγραφής και προγραμματισμού των συσκευών αυτών ποικίλλουν από κατασκευαστή σε κατασκευαστή. Οι κυρίαρχες τεχνολογίες αυτού του τύπου μνήμης είναι δύο: η NOR Flash και η NAND Flash. Οι μνήμες Flash χρησιμοποιούνται σχεδόν σε όλα τα Ε.Σ. για την αποθήκευση δεδομένων και προγράμματος. Η δυνατότητα ενημέρωσής τους από τον κατασκευη ενώ βρίσκονται σε λειτουργία προσδίδει στον τύπο αυτό μνήμης συγκριτικό πλεονέκτημα, αφού, δεν απαιτείται αποσύνδεση της συσκευής και μεταφορά της, αλλά αρκεί η σύνδεσής της στο διαδίκτυο ή σε κάποια συσκευή που φέρει το πρόγραμμα ενημέρωσης. Η πρόσβαση στα δεδομένα της NOR Flash πραγματοποιείται με τυχαίο τρόπο και είναι σχετικά αργή (χαρακτηρίζεται από μεγάλους χρόνους εγγραφής), ενώ

πρόκειται για τεχνολογία υψηλού κόστους συγκριτικά με άλλες μνήμες. Το πλεονέκτημά της ωστόσο, είναι ότι είναι συμβατή με την ανά λέξη (τυπικός αριθμός byte ανά λέξη) διευθυνσιοδότηση και υποστηρίζει άμεση εκτέλεση νέου κώδικα, οπότε και μπορεί να χρησιμοποιηθεί εναλλακτικά άλλου είδους ROM σε οποιοδήποτε κύκλωμα. Η NAND Flash παρέχει μεγαλύτερη πυκνότητα αποθήκευσης, μικρότερο κόστος, λιγότερα καλώδια και πιο μικρούς χρόνους εγγραφής, αλλά το μειονέκτημα είναι ότι οι προσβάσεις γίνονται πάντα σε ομάδες από bit (π.χ. 1024 bit ή 1KByte), γεγονός που αποκλείει την ανεξάρτητη πρόσβαση σε τυχαίο σημείο, αφού τα δεδομένα μεταφέρονται μαζικά κάθε φορά που απαιτείται μια προσπέλαση. Για τον λόγο αυτό, εύκολα μπορεί να συμπεράνει κανείς ότι η λειτουργία της αρμόζει κυρίως για την περίπτωση της αποθήκευσης δεδομένων (αποθηκευτικός χώρος), παρά για την άμεση εκτέλεση, για την οποία αρμόζει η NOR Flash μνήμη. Λόγω του χαμηλού της κόστους, χρησιμοποιείται σε πλήθος Ε.Σ., αλλά με την ιδιαιτερότητα ότι, στην περίπτωση που κώδικας που περιέχει χρήζει εκτέλεσης, την εκτέλεση αυτήν αναλαμβάνει η μνήμη RAM, κατόπιν μεταφοράς του κώδικα σε αυτήν. Για παράδειγμα, όλες οι κάρτες SD που τοποθετούνται στα κινητά τηλέφωνα ή σε Ε.Σ., είναι τεχνολογίας NAND Flash.

3.2.6 Αποθήκευση των Byte

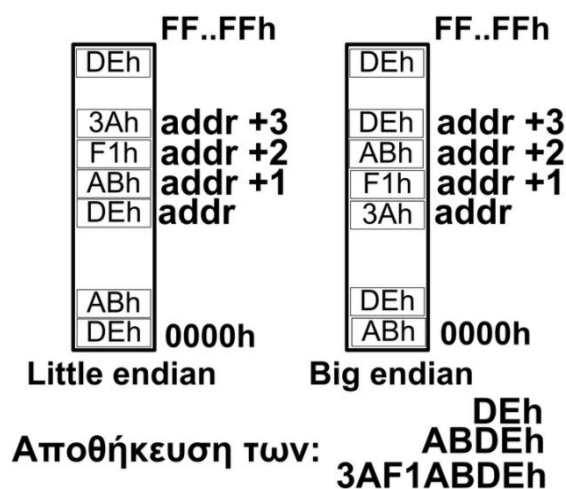
Στις περισσότερες περιπτώσεις, ένας επεξεργαστής διαθέτει μνήμη «εντός» (on-chip) και «εκτός» (off-chip) ολοκληρωμένου κυκλώματος. Στην «εντός» μνήμη συγκαταλέγονται οι «εμφανείς» και οι «αφανείς» καταχωρητές, η μνήμη cache, και η επαναπρογραμματιζόμενη μνήμη μόνο ανάγνωσης. Η «εκτός» μνήμη ομοίως μπορεί να είναι επαναπρογραμματιζόμενη μνήμη μόνο ανάγνωσης ή κάποιο είδος μνήμης τυχαίας προσπέλασης, η οποία ως επί το πλείστο διευθυνσιοδοτείται ανά

λέξη των 8 bit, δηλαδή κάθε διεύθυνση μνήμης έχει χωρητικότητας του 1 Byte. Για την περίπτωση των multi Byte επεξεργαστών, οι οποίοι κατά την αποθήκευση χρησιμοποιούν μεγέθη δεδομένων που ξεπερνούν το ένα Byte, έχουν αναπτυχθεί δυο τεχνικές αποθήκευσης των δεδομένων (καλούνται endianness τεχνικές αποθήκευσης) στην εξωτερική μνήμη, οι οποίες στη βιβλιογραφία αναφέρονται ως «big endian» και «little endian». Ο κάθε επεξεργαστής σχεδιάζεται για να υποστηρίζει τον έναν ή τον άλλο τρόπο αποθήκευσης και αυτό το γεγονός δεν αλλάζει. Υπάρχουν περιπτώσεις που ο επεξεργαστής έχει δυνατότητα να υποστηρίζει και τους δυο τρόπους αποθήκευσης με χρήση επιπρόσθετων κυκλωμάτων, οπότε σε αυτήν την περίπτωση ονομάζεται bi-endian, και μπορεί με την κατάλληλη εντολή κώδικα μηχανής, να αποφασίζεται κατά την εκτέλεση ενός προγράμματος ποιόν από τους δυο ανωτέρω τρόπους θα χρησιμοποιήσει. Σημειώνεται στο σημείο αυτό ότι τεχνική «endianness» είναι κομμάτι της αρχιτεκτονικής αποκλειστικά του επεξεργαστή, γεγονός που καθιστά όλες τις μνήμες συμβατές για χρήση είτε με big είτε με little endian.

Όπως κατέστη σαφές, η endianness αφορά μόνο στην αποθήκευση λέξεων που ξεπερνούν το ένα Byte, διότι στην περίπτωση του ενός μόνο Byte, δεν υπάρχει διαφοροποίηση στην αποθήκευση με χρήση της little ή της big endian αρχιτεκτονικής. Άρα, ο διαφορετικός καθορισμός των θέσεων αποθήκευσης των δεδομένων από την endianness αρχιτεκτονική υφίσταται μόνο στην περίπτωση multi Byte λέξεων. Συγκεκριμένα, η big endian αρχιτεκτονική σε έναν επεξεργαστή υποδηλώνει ότι το μεγαλύτερης σημασίας Byte (MSB, Most Significant Byte) αποθηκεύεται στη μικρότερης προτεραιότητας θέση διεύθυνσης και ακολουθούν διαδοχικά τα επόμενα Byte, έως ότου αποθηκευτεί και το Byte μικρότερης σημασίας (LSB, Least Significant Byte), σε αντίθεση με την little endian αρχιτεκτονική, στην οποία πραγματοποιείται η αντίστροφη διαδικασία

αποθήκευσης, δηλαδή αποθήκευση του LSB στη μικρότερης προτεραιότητας διεύθυνση μνήμης και διαδοχική αποθήκευση Byte έως το MSB.

Στο Σχ.3-8 φαίνεται η αποθήκευση τριών λέξεων διαφορετικού μεγέθους Byte, προκειμένου να αποσαφηνιστεί η διαφορά των δυο αντίθετων τεχνικών αποθήκευσης που αναλύθηκε παραπάνω. Η πρώτη λέξη έχει μέγεθος 1 Byte και είναι η DEh, η δεύτερη έχει μέγεθος 2 Byte και είναι η ABDEh και η τρίτη λέξη μεγέθους 4 Byte είναι η 3AF1ABDEh.



Σχήμα 3-8 Big endian και Little Endian Αρχιτεκτονική [1], [5]

Όπως γίνεται αντιληπτό από το παραπάνω σχήμα, η πρώτη λέξη του ενός Byte αποθηκεύεται σε ορισμένη διεύθυνση (έστω στη FFFFh) ομοιοτρόπως, ανεξαρτήτου της endianness αρχιτεκτονικής. Στην περίπτωση της δεύτερης λέξης, το MSB είναι το ABh, και το LSB είναι το DEh, οπότε στην little endian τεχνική το DEh αποθηκεύεται στη χαμηλότερης προτεραιότητας διεύθυνση, δηλαδή πρώτο, και στην επόμενη διεύθυνση (εδώ συμπίπτει με την υψηλότερης προτεραιότητας διεύθυνση) το ABh, ενώ στη big endian αποθηκεύεται το ABh στη

χαμηλότερης προτεραιότητας διεύθυνση και το DEh στην επόμενη. Για τη λέξη των τεσσάρων Byte 3AF1ABDEh, με MSB το 3Ah και LSB το DEh, και διεύθυνση μνήμης αποθήκευσης την *addr*, σύμφωνα με την little endian το LSB, δηλαδή το DEh, αποθηκεύεται πρώτο και στην επόμενη διεύθυνση (δηλαδή στην *addr+1*) το επόμενο Byte, που είναι το ABh. Ακολούθως, στη διεύθυνση *addr+2* αποθηκεύεται το F1h και τελικά στην διεύθυνση *addr+3* το MSB 3Ah. Η αντίστροφη διαδικασία ακολουθείται στην big endian, όπου το MSB 3Ah αποθηκεύεται πρώτο στην χαμηλότερης προτεραιότητας διεύθυνση, και στη συνέχεια τα υπόλοιπα με αντίστροφη σειρά, F1h, ABh και το DEh στην υψηλότερης προτεραιότητας διεύθυνση.

Κατόπιν της παραπάνω ανάλυσης, μπορεί κανείς να παρατηρήσει ότι στην little endian τεχνική, το LSB αποθηκεύεται στη χαμηλότερης προτεραιότητας διεύθυνση μνήμης, ιδιότητα που την καθιστά εύκολα αντιληπτή από έναν προγραμματιστή, αφού υπάρχει ευθεία αναλογία χαμηλότερης προτεραιότητας θέσης-χαμηλότερης σημασίας λέξης. Αυτό που ίσως «ξενίζει» στην little endian τεχνική είναι το γεγονός ότι το στοιχείο αποθηκεύεται στη μνήμη του υπολογιστή διαφορετικά από ότι διαβάζεται από τον άνθρωπο. Η αποθήκευση της λέξης 3AF1ABDEh στη μνήμη οδηγεί στην αποθήκευση DEABF13Ah στη μνήμη. Ωστόσο, κατά την ανάκληση της λέξης από επεξεργαστή που υποστηρίζει τον τρόπο αποθήκευσης με little endian αρχιτεκτονική, αυτή ανασύρεται ορθά, ανεξάρτητα από τον τρόπο που αποθηκεύονται τα δεδομένα στη μνήμη.

Εναλλακτικά, ένας επεξεργαστής με big endian αρχιτεκτονική που αποθηκεύει την τιμή 3AF1ABDEh στη μνήμη καταλήγει να αποθηκεύσει τελικά την ίδια τιμή 3AF1ABDEh μέσα στο ολοκληρωμένο μνήμης. Κανένα σχέδιο δεν πλεονεκτεί του άλλου, η λειτουργία τους είναι ίδια, είναι δύο διαφορετικοί τρόποι με το ίδιο

αποτέλεσμα. Όταν πραγματοποιείται υψηλού επιπέδου προγραμματισμός σε ένα σύστημα, το endianness έχει μικρή διαφορά. Η μόνη περίπτωση που χρειάζεται κάτι τέτοιο είναι όταν εξετάζονται τα στοιχεία πολλαπλών Byte άμεσα στη μνήμη και απαιτείται η ερμηνεία τους. Εντούτοις, όταν αναπτύσσεται και διορθώνεται το υλικό (δηλαδή σε χαμηλό επίπεδο) σταθερού λογισμικού, συναντώνται πολύ συχνά και έτσι μια κατανόηση big endian και little endian αρχιτεκτονικών από τον σχεδιαστή των Ε.Σ. είναι απαραίτητη.

Ένα άλλο πεδίο στο οποίο χρησιμοποιείται η endianness αρχιτεκτονική πέρα από την απλή μεταφορά δεδομένων (λέξεων) στη μνήμη είναι οι δικτυακές επικοινωνίες, δηλαδή η αποθήκευση δεδομένων οπουδήποτε προβλέπεται κατόπιν αποστολής και η λήψης πληροφοριών μέσω πρωτοκόλλων του διαδικτύου (παραδείγματος χάριν το πρωτόκολλο TCP/IP). Αυτό οφείλεται στο γεγονός, ότι σε περίπτωση που ο αποστολέας και ο παραλήπτης έχουν διαφορετικά endianness, υπάρχει η πιθανότητα να ερμηνεύσουν διαφορετικά τα Byte. Για παράδειγμα, αν ένας υπολογιστής λάβει τη λέξη 3AF1ABDEh και είναι big endian, θα ερμηνεύσει τα δεδομένα ως τον 32bit αριθμό 3AF1ABDEh, ενώ αν είναι little endian θα ερμηνεύσει τα δεδομένα ως DEABF13Ah. Λόγω των ανωτέρω, εξειδικευμένες συναρτήσεις ανασύρονται πριν την έναρξη της αποστολής δεδομένων και μετά το τέλος της λήψης τους, με σκοπό την μετατροπή των δεδομένων σε σειρά αποθήκευσης δικτύου (NBO, network byte order) ώστε τα δεδομένα που στέλνονται να μετατρέπονται κατά τη λήψη από την NBO στην endianness του υπολογιστή. Οι συναρτήσεις αυτές καλούνται ανά δυο, ως ζεύγη κωδικοποίησης/αποκωδικοποίησης τρόπον τινά, όπως στην περίπτωση των συναρτήσεων htons() και ntohs(), οι οποίες περιλαμβάνονται στο πρότυπο IEEE Std 1003.1-2001 ('POSIX.1'). Έτσι, το πρόβλημα της διαφορετικής αρχιτεκτονικής endianness που συχνά εμφανίζεται κατά την αποθήκευση δεδομένων

από αρχεία που μεταφέρονται από/προς διαφορετικά υπολογιστικά συστήματα μπορεί να λυθεί με τη χρήση των συναρτήσεων που προαναφέρθηκαν ή και άλλων παρόμοιων, ακόμα και στην περίπτωση απουσίας δικτυακών επικοινωνιών.

Η endianness είναι πολύ σημαντική στα Ε.Σ., αφού σε αντίθεση με τους επιτραπέζιους υπολογιστές ή διακομιστές, στους οποίους όλα είναι συμβατά με x86 (little endian), υπάρχουν πολλές διαφορετικές αρχιτεκτονικές ή σχεδιασμοί νέων αρχιτεκτονικών και έτσι αυτή η γνώση είναι απαραίτητη σε έναν μηχανικό που εμπλέκεται με αυτά τα συστήματα.

Ως προς τα προτερήματα της κάθε μιας endianness αρχιτεκτονικής ισχύουν τα ακόλουθα: στη little endian, μια λέξη μπορεί να διαβάσει με χρήση διαφορετικού αριθμού bit, για παράδειγμα η λέξη 2C 00 00 00 μπορεί να διαβαστεί είτε ως λέξη μήκους 8 bit (2C), είτε ως λέξη 16 bit (002C), χωρίς καμία αλλαγή στο αποτέλεσμα (σε αυτό στηρίζεται η λειτουργία των μεταγλωττιστών). Από την άλλη, η big endian βοηθάει στο να βρεθεί πόσο μεγάλος είναι ο αριθμός, διαβάζοντας μόνο την πρώτη θέση μνήμης (που φέρει το MSB). Τέλος η little-endian απλοποιεί το υλικό (hardware) σε πράξεις πολλαπλών Byte.

3.2.7 Ευθυγραμμισμένες προσβάσεις στα Δεδομένα

Σε ορισμένους επεξεργαστές απαιτείται ευθυγράμμιση της πρόσβασης στα δεδομένα κατά τις λειτουργίες ανάγνωσης και εγγραφής στη μνήμη. Η ιδιότητα αυτή μπορεί να συσχετιστεί με όσα αναλύθηκαν στην προηγούμενη ενότητα περί αποθήκευσης των δεδομένων, όταν οι λέξεις αποτελούνται από περισσότερα του ενός Byte. Αν η πρόσβαση αφορά μόνο 1 Byte, τότε δεν τίθεται θέμα ευθυγράμμισης. Για κάποιους επεξεργαστές, η ευθυγράμμιση των δεδομένων

αποτελεί σημαντικό στοιχείο της αρχιτεκτονικής τους, οπότε η έλλειψη ευθυγράμμισης κατά την πρόσβαση στη μνήμη μπορεί να οδηγήσει τον επεξεργαστή ακόμα και σε κατάσταση εσφαλμένης εντολής. Οι μη ευθυγραμμισμένες προσβάσεις προκαλούν σημαντική επιβάρυνση στον χρόνο πρόσβασης (διπλασιάζεται) και γι' αυτό όλοι οι σύγχρονοι συμβολομεταφραστές αναδιατάσσουν τον κώδικα ή προσθέτουν στη μνήμη Byte συμπλήρωσης (memory padding) προκειμένου να ευθυγραμμιστούν οι προσβάσεις, όπως περιγράφεται στη συνέχεια.

Η αναντιστοιχία του μήκους των λέξεων που χρησιμοποιούνται από επεξεργαστές διαφορετικών αρχιτεκτονικών και δόμησης αναπόφευκτα εισάγει το πρόβλημα της μη ευθυγραμμισμένης πρόσβασης σε δεδομένα που μεταφέρονται από και προς τη μνήμη και τις υπόλοιπες δομικές μονάδες του συστήματος. Αυτό το εγγενές μήκος λέξης χαρακτηρίζει τον επεξεργαστή (NBW, Native Bit Width). Μέχρι σήμερα μπορεί κάποιος να χρησιμοποιήσει στα Ε.Σ. επεξεργαστές με 8, 16, 32 ή 64 bit. Ο αριθμός αυτός των bit ανά λέξη καθορίζει και την ικανότητα μεταφοράς δεδομένων που διαθέτει το σύνολο των διαύλων του επεξεργαστή (εσωτερικών και εξωτερικών). Εύκολα λοιπόν προκύπτει το συμπέρασμα ότι η μεγαλύτερη συχνότητα πρόσβασης (αντιστοιχεί στον μικρότερο χρόνο εξυπηρέτησης) επιτυγχάνεται όταν μια μεταφορά απαιτεί μήκος λέξεων ίσο με του εγγενές μήκους λέξεων του επεξεργαστή. Με άλλα λόγια, για 8 bit επεξεργαστές η βέλτιστη εξυπηρέτηση επιτυγχάνεται με μεταφορές των 8 bit, ενώ μεταφορές των 16 ή 32 bit ελαττώνουν την ταχύτητα της εξυπηρέτησης.

Η μέγιστη ταχύτητα πρόσβασης στη μνήμη επιτυγχάνεται όταν οι επεξεργαστές διαβάζουν ή γράφουν με μήκος λέξης τόσο όσο το εγγενές μήκος λέξης του επεξεργαστή. Οι διευθύνσεις μνήμης των επεξεργαστών οργανώνονται σε

υποομάδες μεγέθους λέξεων (Byte), που αντιστοιχούν στο εγγενές μήκους λέξεων του επεξεργαστή. Η παραπάνω αρχή για έναν 32 bit επεξεργαστή υποδηλώνει ομαδοποίηση των διευθύνσεων της μνήμης ανά 4 λέξεις (Byte), κατά την οποία δε λαμβάνονται υπόψη τα χαμηλότερης σημασίας bit βάσει των οποίων διευθυνσιοδοτούνται οι λέξεις σε κάθε ομάδα. Έτσι, για τον 32 bit επεξεργαστή, 2 είναι τα bit χαμηλότερης σημασίας που διευθυνσιοδοτούν τις 4 λέξεις της υποομάδας, οπότε και αγνοούνται. Αντίστοιχα, για τον 64 bit επεξεργαστή, τα bit χαμηλότερης σημασίας της διεύθυνσης μνήμης που διευθυνσιοδοτούν τις 8 λέξεις της υποομάδας και αγνοούνται είναι 4.

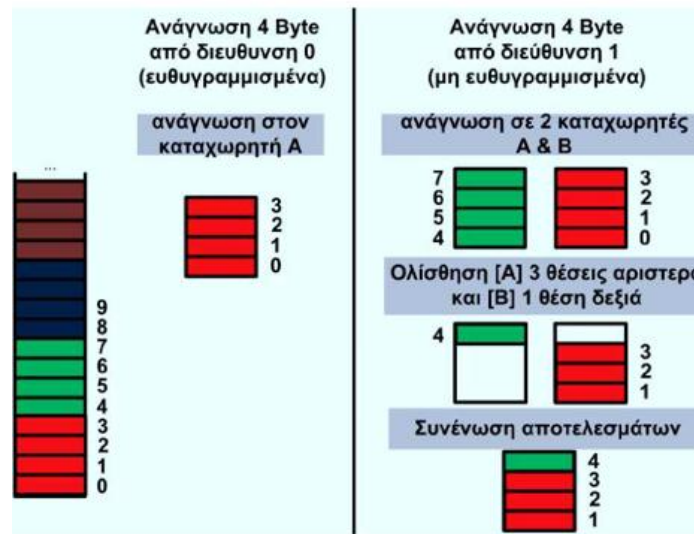
Σε ορισμένα συστήματα, προκειμένου να μειωθεί ο αριθμός των συνδέσεων με καλώδια και να ελαττωθεί η κατανάλωση ενέργειας, η τεχνική της ομαδοποίησης αξιοποιείται για την απαλοιφή των χαμηλότερης σημασίας bit από τον εξωτερικό δίαυλο διευθύνσεων, θεωρώντας τα εξ' ορισμού ως βάση της υποομάδας (0 x αριθμός bit χαμηλότερης σημασίας). Έτσι σε ένα 32 bit σύστημα, αντί για 32 bit χρησιμοποιούνται 30 bit ή μπορούν να χρησιμοποιηθούν τα ίδια bit για να διευθυνσιοδοτήσουν 4 φορές περισσότερη μνήμη (π.χ. για 32 bit να διευθυνσιοδοτήσουν 34 bit). Σύμφωνα με τα παραπάνω, σε έναν 32 bit επεξεργαστή, η αναπαράσταση της διεύθυνσης ABF1h στο δυαδικό σύστημα είναι η 1010101111110001 και η υποομάδα του συστήματος είναι η 10101011111100xx, δηλαδή από 1010101111110000 (ABF0h) έως 1010101111110011 (ABF3h), ενώ σε ένα 64 bit σύστημα η υποομάδα είναι η 101010111111xxxx, δηλαδή από 1010101111110000 (ABF0h) έως 1010101111111111 (ABFFh). Η μικρότερη διεύθυνση καλείται βάση της υποομάδας ενώ η μεγαλύτερη κορυφή της.

Για να θεωρηθεί μια πρόσβαση περισσότερων του ενός Byte στη μνήμη ευθυγραμμισμένη, πρέπει αυτή να πραγματοποιείται στη βάση της υποομάδας και

όχι σε κάποια ενδιάμεση διεύθυνσή της. Σε συνέχεια του προηγούμενου παραδείγματος του μια πρόσβαση για την ανάγνωση των 4 λέξεων της διεύθυνση ABF0h (αποτελεί τη βάση της υποομάδας των λέξεων ABF0h - ABF3h) θεωρείται ευθυγραμμισμένη, γιατί με μια μονάχα πρόσβαση μπορούν να μεταφερθούν και οι 4 λέξεις ABF0h - ABF3h. Μια μη ευθυγραμμισμένη πρόσβαση θα μπορούσε να χαρακτηριστεί αυτή σε μια ενδιάμεση διεύθυνση της υποομάδας, όπως στην περίπτωση της ανάγνωσης 4 λέξεων στη διεύθυνση ABF1h, εφόσον ορισμένες λέξεις συγκαταλέγονται σε μια υποομάδα (ABF1h, ABF2h, ABF3h) και απαιτείται μια πρόσβαση για αυτές και ορισμένες σε διαφορετική (συγκεκριμένα η τέταρτη λέξη, ABF4h, ανήκει στην επόμενη υποομάδα), οπότε απαιτείται μια επιπρόσθετη πρόσβαση γι' αυτήν. Το αποτέλεσμα της παραπάνω διαδικασίας είναι ο υποδιπλασιασμός της ταχύτητας πρόσβασης.

Για την υποστήριξη των ευθυγραμμισμένων προσβάσεων μνήμης απαιτείται η εισαγωγή συγκεκριμένων στοιχείων στο κύκλωμα του επεξεργαστή, τα οποία να είναι υπεύθυνα για την εξασφάλισή της. Τα στοιχεία αυτά είναι ορισμένοι προσωρινοί καταχωρητές (στους οποίους ο χρήστης δεν έχει πρόσβαση), στους οποίους εφαρμόζεται ολίσθηση μέχρι ενός οριακού αριθμού θέσεων (έως και 3 θέσεις για 32 bit επεξεργαστή), και ένας καταχωρητής στον οποίο θα συνενωθούν τα δεδομένα των επιμέρους καταχωρητών. Στο Σχ.3-9 παρουσιάζεται ένα τέτοιο παράδειγμα, όπου φαίνεται ότι η ευθυγραμμισμένη πρόσβαση γίνεται άμεσα και η τιμή αποθηκεύεται στον καταχωρητή, ενώ η μη ευθυγραμμισμένη υπόκειται σε κάποια επεξεργασία προκειμένου να ολοκληρωθεί. Η απαίτηση της επιπρόσθετης επεξεργασίας μαζί με το μειονέκτημα του διπλασιασμού του χρόνου πρόσβασης στη μνήμη, εκτός του ότι καθυστερεί τον επεξεργαστή, δεσμεύει και έναν αριθμό τρανζίστορ για την υλοποίηση των ειδικών κυκλωμάτων. Για αυτόν τον λόγο, αρκετοί επεξεργαστές δεν υποστηρίζουν μη ευθυγραμμισμένες προσβάσεις, αφού

στις αρχιτέκτονες αυτών επιλέγεται να αξιοποιηθεί αλλού αυτός ο αριθμός των τρανζίστορ.



Σχήμα 3-9 Παράδειγμα μη ευθυγραμμισμένης πρόσβασης στη μνήμη [1]

Πέρα από την αύξηση του χρόνου πρόσβασης που οδηγεί στην απαίτηση μεγαλύτερης χωρητικότητας (πρόσθετο υλικό), η εφαρμογή μη ευθυγραμμισμένων προσβάσεων παρεμποδίζει και τις λειτουργίες συγχρονισμού του ρολογιού του επεξεργαστή με τις εντολές του συστήματος. Όλοι οι σύγχρονοι επεξεργαστές που προορίζονται για λειτουργικά συστήματα με πολλαπλές διεργασίες έχουν δομές συγχρονισμού ανάμεσα σε νήματα και διεργασίες. Αυτές οι δομές απαιτούν μεμονωμένες λειτουργίες ανάγνωσης ή εγγραφής στη μνήμη, δηλαδή λειτουργίες που εκτελούνται ενιαία και αδιάσπαστα σε όσο το δυνατόν ελάχιστους κύκλους. Οι περισσότεροι του ενός κύκλοι πρόσβασης που απαιτούνται όταν η πρόσβαση δεν είναι ευθυγραμμισμένη μπορούν να οδηγήσουν σε εκτέλεση ανεπιθύμητης

διακοπής (interrupt) σε μη προγραμματισμένη φάση λειτουργίας με κίνδυνο απώλειας του συγχρονισμού. Για αυτόν τον λόγο υπάρχει η απαίτηση τουλάχιστον οι δομές συγχρονισμού να είναι ευθυγραμμισμένες, ακόμα και στα συστήματα που υποστηρίζουν μη ευθυγραμμισμένες προσβάσεις.

Γενικά, οι μη ευθυγραμμισμένες προσβάσεις πρέπει να αποφεύγονται, κι έτσι, τα τελευταία χρόνια όλοι οι συμβολομεταφραστές, απουσία ευθυγράμμισης, προγραμματίζονται να προσθέτουν έξτρα Byte(s) ώστε να ευθυγραμμιστούν τα δεδομένα.

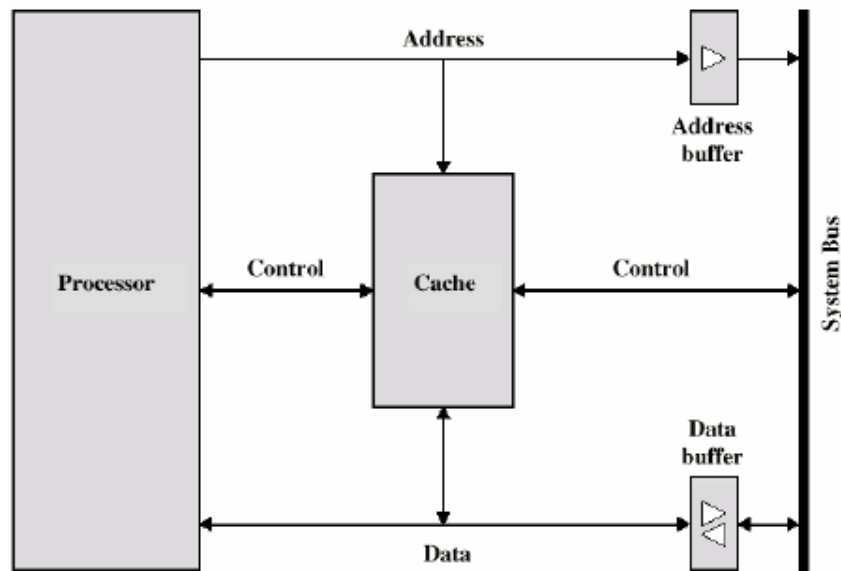
3.2.8 Κρυφή Μνήμη [1], [5]

Κάθε πρόσβαση στην εξωτερική μνήμη επιφέρει μια μεγαλύτερη ενεργειακή και χρονική επιβάρυνση σε σχέση με την πρόσβαση σε μνήμη εντός του ολοκληρωμένου κυκλώματος. Επίσης, τα περισσότερα προγράμματα παρουσιάζουν μια χρονική και χωρική τοπικότητα, οι οποίες μπορούν με κατάλληλη διαχείριση να

βελτιώσουν την απόδοση του συστήματος. Η χρονική τοπικότητα στις προσβάσεις ερμηνεύεται ως εξής: υπάρχει αυξημένη πιθανότητα ένας επεξεργαστής να κληθεί να προσπελάσει την ίδια διεύθυνση μνήμης πολλές φορές κατόπιν σύντομου χρονικού διαστήματος. Η χωρική τοπικότητα στις προσβάσεις ερμηνεύεται ως εξής: υπάρχει αυξημένη πιθανότητα ένας επεξεργαστής να κληθεί να προσπελάσει γειτονικές διευθύνσεις της μνήμης που προσπέλασε το αμέσως προηγούμενο χρονικό διάστημα. Αν ληφθούν υπόψη οι δυο παραπάνω αρχές, τότε αποδεικνύεται συμφέρουσα από άποψη ενεργειακού και χρονικού κέρδους η αντιγραφή των δεδομένων της εξωτερικής μνήμης και των γειτονικών διευθύνσεων κατά την πρόσβαση σε κάποια εσωτερική μνήμη, ώστε να είναι άμεσα διαθέσιμες για επερχόμενη προσπέλαση. Στην περίπτωση δε, που μια μνήμη

αυτής της λογικής «στήνεται» με τρόπο τέτοιο ώστε να λειτουργεί μην παρεμβαίνοντας στο πρόγραμμα (η λειτουργία χαρακτηρίζεται ως κρυφή), τότε διευκολύνεται άμεσα η εκτέλεση των υπαρχόντων προγραμμάτων όπως εξηγήθηκε παραπάνω. Μια μνήμη αυτού του είδους ονομάζεται «κρυφή μνήμη» (cache memory) και ουσιαστικά γεφυρώνει το χάσμα ανάμεσα στην πολύ γρήγορη προσπέλαση ενός μικρού όγκου δεδομένων και τον μεγάλο χρόνο που απαιτείται για την πρόσβαση σε όλο και μεγαλύτερο όγκο δεδομένων. Ποικίλες δομές ιεραρχίας από κρυφές μνήμες χρησιμοποιούνται απαραίτητα σε όλους τους σύγχρονους επεξεργαστές.

Η κρυφή μνήμη λειτουργεί ως εξής: Όταν είναι επιθυμητό ο επεξεργαστής να προσπελάσει (να διαβάσει ή εγγράψει) μια διεύθυνση κύριας μνήμης, πρώτα ελέγχεται η ύπαρξη αντίγραφου των δεδομένων της τοποθεσίας αυτής στην κρυφή μνήμη. Η πραγματοποίηση μεταφοράς στην κρυφή μνήμη καλείται επιτυχία κρυφής μνήμης (η ορολογία αναφέρεται σε cache hit) και επιτρέπει άμεση ολοκλήρωση της πρόσβασης. Αν όμως δεν έχει γίνει αντιγραφή, τότε υπάρχει αστοχία κρυφής μνήμης (cache miss) και πρέπει πρώτα να διαβαστεί η διεύθυνση (και ίσως και κάποιες γειτονικές διευθύνσεις) και να μεταφερθούν τα δεδομένα μέσα στην κρυφή μνήμη. Με την ολοκλήρωση της αντιγραφής, καθίσταται δυνατή η ανάγνωση των δεδομένων της κρυφής μνήμης από τον επεξεργαστή. Τα παραπάνω αναφέρονται σε προσπελάσεις ανάγνωσης και όχι εγγραφής (δεν υπάρχει νόημα να επαναχρησιμοποιούνται δεδομένα που γράφονται συνεχώς).



Σχήμα 3-10 Τοπική οργάνωση της κρυφής μνήμης [5]

Η περιγραφή αυτή της λειτουργίας της κρυφής μνήμης οδηγεί σε διάφορες επιλογές σχεδιασμού κρυφής μνήμης, όπως η χαρτογράφηση κρυφής μνήμης, πολιτική αντικατάστασης της κρυφής μνήμης και τεχνικές εγγραφής στην κρυφή μνήμη. Αυτές οι επιλογές σχεδιασμού μπορεί να έχουν πολλαπλές επιδράσεις στο κόστος του συστήματος, την απόδοσή του καθώς και στην κατανάλωση ενέργειας, συνεπώς πρέπει να αξιολογηθούν προσεκτικά για κάθε εφαρμογή ενός Ε.Σ.

3.3 Ο επεξεργαστής ενός Ε.Σ.

Στα Ε.Σ. υπάρχει μια μεγάλη ποικιλία υλοποιήσεων, με τις απαιτήσεις του εκάστοτε συστήματος να καθορίζουν τον τύπο του επεξεργαστή που θα χρησιμοποιηθεί. Στο εμπόριο διατίθενται επεξεργαστές γενικής χρήσης, όπως τα

περισσότερα μοντέλα της Intel ή της AMD, ειδικοί επεξεργαστές ψηφιακής επεξεργασίας σημάτων (επεξεργαστές της Texas Instruments), επεξεργαστές εκπαιδευτικού σκοπού(π.χ. AVR ATMEGA που χρησιμοποιούν οι μικροελεγκτές Arduino), σύνθετοι και απλοί 32 bit επεξεργαστές εξειδικευμένων εφαρμογών, επεξεργαστές μαλακού πυρήνα (soft-cores) που χρησιμοποιούνται για την υλοποίηση της λογικής των FPGA (Field Programmable Gate Array ή συστοιχία επιτόπια προγραμματιζόμενων πυλών), καθώς και ειδικής διαμόρφωσης κυκλωμάτων χωρίς επεξεργαστή, τα οποία κατασκευάζονται για την υλοποίηση μια συγκεκριμένης λειτουργίας με χρήση μηχανής πεπερασμένων καταστάσεων (finite state machine). Επισημαίνεται ότι οι επεξεργαστές των Ε.Σ., επειδή κατασκευάζονται για συγκεκριμένη λειτουργία, ως επί το πλείστον δεν μπορούν να λειτουργήσουν σε διαφορετικά Ε.Σ., εκτός κι αν πρόκειται για συστήματα παρόμοιας λειτουργίας. Αυτό αποτελεί μια από τις θεμελιώδεις διαφοροποιήσεις των Ε.Σ. από τους υπολογιστές γενικού σκοπού (σταθεροί και φορητοί προσωπικοί υπολογιστές) οι οποίοι είναι συμβατοί μεταξύ τους και συμβαδίζουν με τις προδιαγραφές της αρχιτεκτονικής επεξεργαστών x86 ή x64, και κατ' επέκταση μπορούν να εκτελέσουν το ίδιο πρόγραμμα χωρίς να απαιτούνται τροποποιήσεις.

3.4 Αρχιτεκτονική αποθηκευμένου προγράμματος

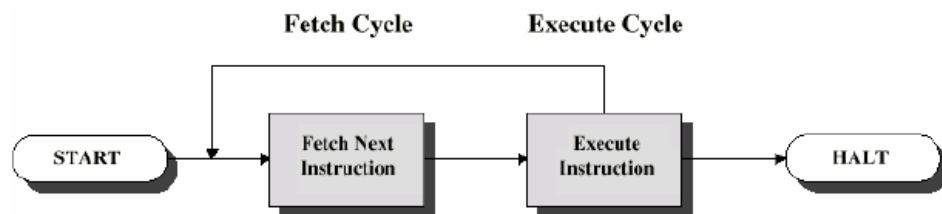
Σε αυτή την ενότητα περιγράφεται η ισχύουσα αρχιτεκτονική στα προγραμματιζόμενα υπολογιστικά συστήματα, που είναι ίδια είτε πρόκειται για Ε.Σ., είτε για οποιοδήποτε άλλο. Η αρχιτεκτονική αυτή ονομάζεται αποθηκευμένου προγράμματος (stored-program) γιατί το πρόγραμμα βρίσκεται αποθηκευμένο μόνιμα (π.χ. σε έναν σκληρό δίσκο ή σε μια μνήμη Flash), μεταβαίνει στη μνήμη τυχαίας προσπέλασης και στη συνέχεια ο επεξεργαστής λαμβάνει σταδιακά τις εντολές του προγράμματος και τις εκτελεί. Μόλις

εκτελείται μια εντολή, τότε προσκομίζεται η επόμενη και ούτω καθεξής.

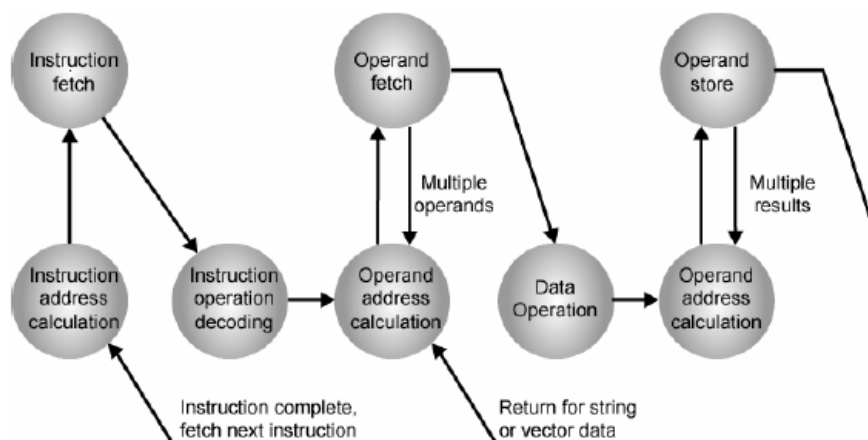
Από τα παραπάνω φαίνεται ότι ο επεξεργαστής εκτελεί συνεχώς έναν ατέρμων βρόχο, που αποτελείται από τα βήματα της προσκόμισης της εντολής και της εκτέλεσης. Στην πράξη, ο επεξεργαστής εκτελεί συνεχώς τέσσερις λειτουργικούς κύκλους:

1. τον κύκλο προσκόμισης της εντολής (fetch cycle),
2. την αποκωδικοποίηση της εντολής (instruction decode),
3. τον κύκλο εκτέλεσης της εντολής (execute cycle) και
4. την αποθήκευση των αποτελεσμάτων (store)

Η εντολή που προορίζεται να ξεκινήσει την εκτέλεση και βρίσκεται στην εξωτερική μνήμη προσκομίζεται από τον επεξεργαστή σε έναν εσωτερικό καταχωρητή εντολής (Instruction Register). Η εντολή αυτή είναι κωδικοποιημένη με τα λογικά bit 1 και 0 σε συγκεκριμένες θέσεις.



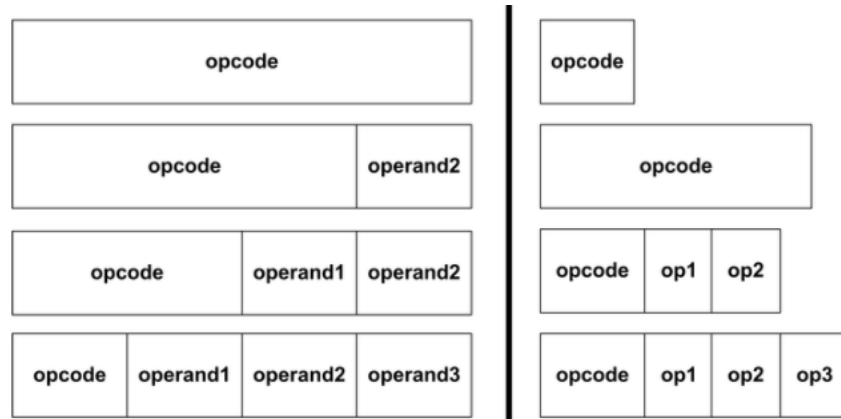
Σχήμα 3-11 Ο βασικός κύκλος εντολών [5]



Σχήμα 3-12 Διάγραμμα καταστάσεων κύκλου εντολής [5]

Οι επεξεργαστές χωρίζονται σε 2 κατηγορίες ως προς το μήκος των εντολών που υποστηρίζουν (Σχήμα 3-12), σε αυτούς με εντολές σταθερού μήκους bit, στους οποίους όλες οι εντολές έχουν το ίδιο μήκος (π.χ. 32 bit) και σε αυτούς με εντολές μεταβλητού μήκους bit, στους οποίους όλες οι εντολές δεν έχουν το ίδιο μήκος, άλλες εντολές είναι 1 Byte, άλλες εντολές 2 Byte, κτλ. Οι εντολές σταθερού αριθμού bit αποκωδικοποιούνται ευκολότερα και συνεπώς καταλαμβάνουν μικρότερο επεξεργαστικό χρόνο για αυτή τους την λειτουργία. Γι' αυτό και η πλειονότητα των επεξεργαστών που χρησιμοποιούνται στα Ε.Σ. έχει χρησιμοποιεί εντολές σταθερού μήκους. Το συγκριτικό πλεονέκτημα των εντολών μεταβλητού μήκους έγκειται στο ότι παρέχουν ευελιξία και προσφέρονται για ποικίλες προσαρμογές στις ανάγκες της εκάστοτε εφαρμογής κατά τον προγραμματισμό, αφού επιτρέπουν την ανάπτυξη πολύπλοκης λειτουργίας κώδικα με χρήση μιας μόνο εντολής (π.χ. έλεγχος σε μια περιοχή μνήμης για την ύπαρξη μιας αλφαριθμητικής ακολουθίας-string). Αντίστοιχες πολύπλοκες λειτουργίες δε θα μπορούσαν να υλοποιηθούν μια εντολή σταθερού αριθμού bit, αλλά θα απαιτούσαν πολύ περισσότερες εντολές, άρα και προσκομίσεις στους στη μνήμη. Ωστόσο, αξίζει να ληφθεί υπόψη ότι κατά τη χρήση πολύπλοκων εντολών, ο

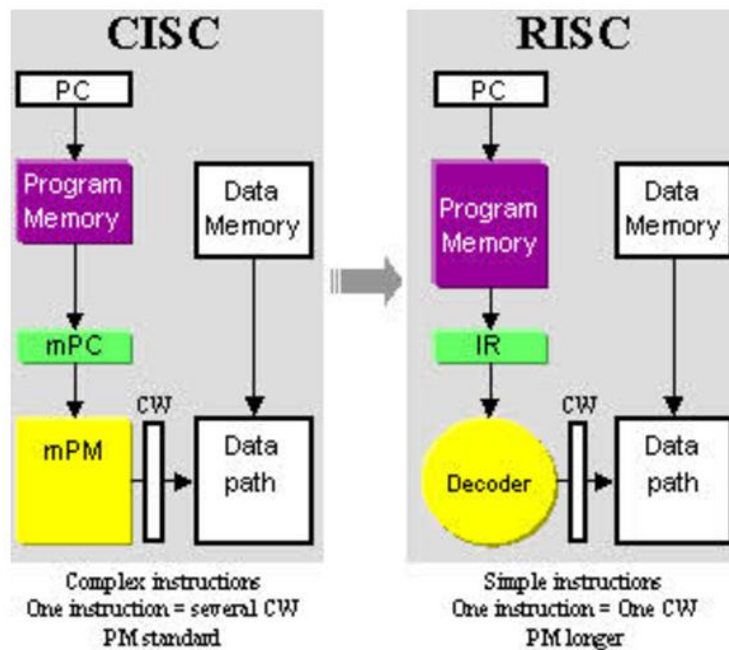
απαιτούμενος χρόνος εκτέλεσης κάθε εντολής δεν μπορεί πάντα να υπολογιστεί με ακρίβεια: κάποιες εντολές θα απαιτήσουν 1 κύκλο, κάποιες 10 κύκλους, κτλ., σε αντιδιαστολή με τις εντολές σταθερού μήκους που απαιτούν στην πλειονότητά τους χρόνο εκτέλεσης ενός κύκλου ρολογιού.



Σχήμα 3-13 Οι εντολές σταθερού μήκους bit (αριστερά) και οι εντολές μεταβλητού μήκους bit (δεξιά), με διάφορες μορφές ως προς τις παραμέτρους [1]

Οι ιδιότητες αυτές (σταθερό μήκος εντολής, χρόνος εκτέλεσης εντολής ίσο με 1 κύκλο ρολογιού, λίγες εντολές και όχι πολύπλοκες, απλοποιημένη λογική αποκωδικοποίησης) χαρακτηρίζουν τους υπολογιστές της αρχιτεκτονικής RISC (Reduced Instruction Set Computing, επεξεργαστές με μειωμένο σύνολο εντολών), όπως είναι οι επεξεργαστές ARM και MIPS. Εναλλακτικά, υπάρχουν οι επεξεργαστές πολύπλοκου σετ εντολών (CISC, Complex Instruction Set Computing), στους οποίους περιλαμβάνονται κυρίως οι οικογένειες των επεξεργαστών της αρχιτεκτονικής x86 και x64. Οι σημαντικότερες διαφοροποιήσεις των δυο παραπάνω τύπων εντολών μπορούν να συνοψιστούν ως εξής: οι εντολές RISC εκτελούνται σε έναν κύκλο λειτουργίας, όλες οι παράμετροί τους αποθηκεύονται σε καταχωρητές, σε αντίθεση με τους CISC στους οποίους

χρησιμοποιούνται εντολές που περιλαμβάνουν απευθείας πράξη πάνω σε δεδομένα θέσης μνήμης και επιπλέον τα προγράμματα RISC αρχεία μεγαλύτερης έκτασης και μεγέθους, ως συνέπεια των πολλών εντολών που απαιτούν για την ολοκλήρωσή τους. Σε παλαιότερες δεκαετίες, το σύνολο των υπολογιστών χρησιμοποιούσε CISC εντολές. Σαφώς, η ανάπτυξη των Ε.Σ., οδήγησε τους σημερινούς επεξεργαστές στη χρήση εντολών RISC και μάλιστα, ακόμα και οι επεξεργαστές CISC εντολών (χαρακτηριστικό παράδειγμα αποτελεί η οικογένεια επεξεργαστών Intel x86) αποκωδικοποιούνται πλέον σε μικροεντολές RISC. Αξίζει να σημειωθεί στο σημείο αυτό ότι σε ορισμένους RISC επεξεργαστές τελευταίας γενιάς έχουν ενσωματωθεί κάποιες εντολές τύπου CISC, για τη διευκόλυνση ορισμένων λειτουργιών. Ασφαλώς, αυτό δεν αναιρεί το γεγονός ότι εξακολουθούν να είναι αρχιτεκτονικής RISC.



Σχήμα 3-14 Αρχιτεκτονικές CISC και RISC [14]

Εκτός από το μήκος των εντολών, ένα άλλο στοιχείο είναι ο αριθμός των

παραμέτρων που δέχεται μια εντολή. Η σύνταξη των εντολών και η ονομασία των καταχωρητών χαρακτηρίζουν κάθε επεξεργαστή. Υπάρχουν γενικά εντολές που δε παίρνουν καμία παράμετρο και άλλες που παίρνουν 1 ή 2 ή 3 παραμέτρους.

3.4.1 Κώδικας μηχανής

Όλα τα προγράμματα που εξετάζει ένας επεξεργαστής αποθηκεύονται ως ψηφία 1 και 0 στη μνήμη του συστήματος, οπότε απαιτείται μια λογική αποκωδικοποίηση μιας δυαδικής ακολουθίας (λέξης), έστω της ακολουθίας 01101011. Αυτή η λέξη θα μπορούσε να είναι εντολή προς τον επεξεργαστή (κώδικας μηχανής, assembly) ή δεδομένα (ένας χαρακτήρας ASCII ή μια ακέραια τιμή χωρίς πρόσημο (unsigned integer) ή μια ακέραια τιμή με πρόσημο (signed integer) ή τμήμα μιας μεγαλύτερης λέξης). Η εκάστοτε σημασία της τιμής σχετίζεται με το πού βρίσκεται αποθηκευμένη. Αν βρίσκεται σε περιοχή μνήμης που χαρακτηρίζεται ως κώδικας (code), τότε θα εκτελεστεί, ενώ αν βρίσκεται σε κάποιο κομμάτι μνήμης που χαρακτηρίζεται ως δεδομένο, θα αντιμετωπιστεί ανάλογα. Η παραπάνω λειτουργία αφήνει το περιθώριο σε κακόβουλους χρήστες/εισβολείς να τοποθετούν κωδικοποιημένες εντολές στη μνήμη του συστήματος, οι οποίες καθοδηγούν τον επεξεργαστή στην εκτέλεση του κώδικα που έχουν ενσωματώσει για ίδιους σκοπούς.

Οι κώδικες μηχανής διαιρούνται σε πεδία, τα οποία κατά γενική σύμβαση χαρακτηρίζονται ως opcode (operational code, κώδικας πράξης) και operand(s) (παραμέτροι) και κάθε επεξεργαστής ή οικογένεια επεξεργαστών αναγνωρίζει συγκεκριμένο κώδικα μηχανής. Οι διάφοροι κώδικες μηχανής είναι πολύπλοκοι στην κατανόηση και τη γραφή, γι' αυτό και χρησιμοποιείται μια σημειολογική γλώσσα που γεφυρώνει το χάσμα ανάμεσα σε υπολογιστή και άνθρωπο, η

αποκαλούμενη συμβολική γλώσσα (Assembly). Προκειμένου να προγραμματιστεί ένας οποιοδήποτε επεξεργαστής, μπορεί να χρησιμοποιηθεί μια γλώσσα υψηλού επιπέδου, όπως η C, που μεταγλωττίζεται (compile) από τα κατάλληλα εργαλεία (compiler και debuggers), δημιουργώντας τη συμβολική γλώσσα (assembly) που μεταφράζει τη γλώσσα του προγραμματιστή σε τον κώδικα μηχανής, (ακολουθία bit 0 και 1).

Παρότι η μνήμη του συστήματος χρησιμοποιεί το δυαδικό σύστημα για την αποθήκευση των τιμών, έχει επικρατήσει η απεικόνισή τους στο δεκαεξαδικό σύστημα (γλώσσα μηχανής), ώστε να επιτυγχάνεται πιο συμπαγής μορφή. Σε κάθε περίπτωση, ένας επεξεργαστής μπορεί να προγραμματιστεί απευθείας σε γλώσσα συμβόλων. Στο Σχ.3-15 παρουσιάζεται τμήμα προγράμματος σε συμβολική γλώσσα και στον αντίστοιχο κώδικα μηχανής. Κάθε συμβολική εντολή αναπαρίσταται από έναν αριθμό Byte που τοποθετούνται σε διαδοχικές θέσεις μνήμης.

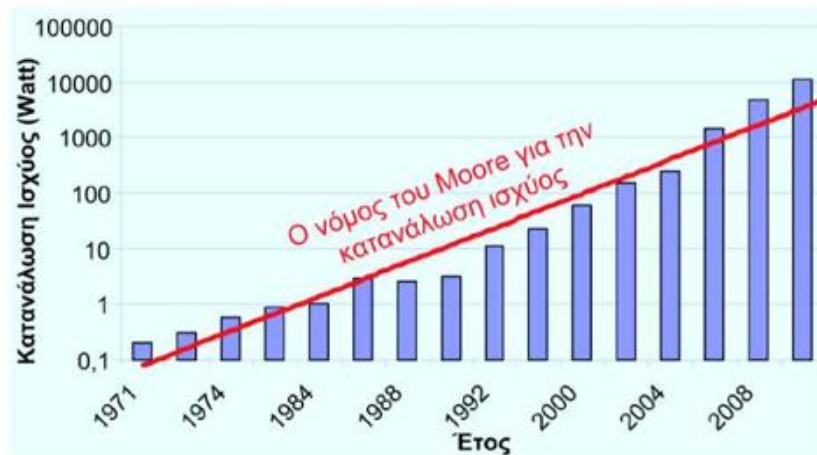
0010: B8 00 00	ARXH: MOV AX, DEDOMENA
0013: 8E D8	MOV DS, AX
⋮	
0015: BA 00 00	LEA DX, MSG
0018: B4 09	MOV AH, 9
001A: CD 21	INT 21H
⋮	
001C: B4 4C	MOV AH, 4CH
001E: CD 21	INT 21H
(α)	(β)
(β)	(γ)

Σχήμα 3-15 Ο κώδικας μηχανής που έχει προέλθει από τη συμβολική γλώσσα (γ), εκφρασμένος στο δεκαεξαδικό σύστημα (β), τοποθετείται σε διαδοχικές διευθύνσεις μνήμης (α) [1]

3.4.2 Απαιτήσεις μείωσης της κατανάλωσης ενέργειας του συστήματος

Η τάση της τεχνολογίας παλαιότερων δεκαετιών ήταν η επίτευξη μέγιστης επίδοσης, με ταυτόχρονη ελαχιστοποίηση του κόστους. Έτσι, έδιναν ιδιαίτερη έμφαση στην ανάπτυξη προϊόντων ολοένα και πιο «γρήγορα», παραμελώντας τελείως την κατανάλωση ενέργειας. Σήμερα, όμως, η κατανάλωση ενέργειας είναι ο βασικός παράγοντας που απασχολεί την τεχνολογική κοινότητα και προς αυτήν την κατεύθυνση κινείται -της μείωσης της κατανάλωσης- κινούνται όλα τα στάδια του σχεδιασμού των υπολογιστικών συστημάτων. Έτσι, η κατανάλωση ενέργειας ενός συστήματος (ή επεξεργαστή), αναφέρεται πάντα μαζί με τα υπόλοιπα χαρακτηριστικά του.

Οι επιδόσεις των σημερινών επεξεργαστών αυξάνονται με ραγδαίους ρυθμούς, γεγονός που συνεπάγεται την εκθετική αύξηση της κατανάλωσης ενέργειας. Αν και αρχικά οι μικροεπεξεργαστές κατανάλωναν λίγα Watt/hour, σήμερα υπάρχουν επεξεργαστές που έχουν κατανάλωση ισχύος πάνω από 80 Watt. Έτσι, πλέον η χρήση συστημάτων ψύξης είναι επιτακτική, η οποία γίνεται ολοένα πιο δύσκολη και ακριβή. Αν συνεχιστεί ο ρυθμός αυτός αύξησης της κατανάλωσης, σε λίγα χρόνια οι καταναλώσεις θα φτάνουν τις εκατοντάδες Watt/hour, που είναι απαγορευτικό για τα περισσότερα συστήματα.



Σχήμα 3-16: Ο νόμος του Moore, σύμφωνα με τον οποίο η κατανάλωση ενέργειας των συστημάτων αυξάνει εκθετικά με το ημερολογιακό έτος [5]

Προφανώς, η λύση στο πρόβλημα της κατανάλωσης ισχύος με μείωση των επιδόσεων δεν είναι αποτελεσματική. Οι χρήστες των φορητών συσκευών απαιτούν την εκτέλεση των εφαρμογών τους με όσο το δυνατό μικρότερη κατανάλωση ισχύος, και όσο γίνεται πιο γρήγορα. Όταν δεν ικανοποιούνται οι επεξεργαστικές απαιτήσεις, ο χρήστης αντιμετωπίζει προβλήματα τύπου διακοπής της αναπαραγωγής ήχου και εικόνας, απώλεια σήματος ή καθυστέρηση φόρτωσης δεδομένων κ.ά.

Ωστόσο, η μεγάλη κατανάλωση ισχύος δεν είναι το μόνο πρόβλημα στις φορητές συσκευές. Η κατανάλωση ισχύος συνδέεται άμεσα με το κόστος λειτουργίας ψύξης. Το κόστος ενσωμάτωσης της μονάδας ψύξης στο σύστημα σε ορισμένες περιπτώσεις είναι αρκετά υψηλό και εξαρτάται από το μέγεθος και την κατανάλωση του στοιχείου του οποίου η αναπτυσσόμενη θερμότητα πρέπει να αποβληθεί στο περιβάλλον. Επιπρόσθετα του αυξημένου κόστους της, μια μεγάλου μεγέθους μονάδα ψύξης είναι συνήθως θορυβώδης, λόγω των τριβών που αναπτύσσονται στα μηχανικά της μέρη. Συνεπώς, υπάρχει ένα συγκριτικό

πλεονέκτημα στη μείωση της κατανάλωσης ισχύος, τόσο για τη μείωση του κόστους όσο και για τη μείωση του θόρυβου που παράγει η συσκευή.

Ένα άλλο εξίσου σημαντικό με τα παραπάνω είναι το θέμα της αξιοπιστίας (reliability). Οι θερμοκρασίες που αναπτύσσονται σε συστήματα υψηλής ενεργειακής κατανάλωσης είναι κι αυτές υψηλές. Οι υψηλές θερμοκρασίες όμως επιδρούν αρνητικά στα κυκλώματα πολύ μεγάλης κλίμακας ολοκλήρωσης (VLSI , Very Large Scale Integration) και οδηγούν σε διάφορες καταστάσεις αστοχίας. Επιπλέον, η διάρκεια ζωής ενός ολοκληρωμένου κυκλώματος είναι αντιστρόφως ανάλογη με την τιμή της θερμοκρασίας στην οποία λειτουργεί. Εν ολίγοις, η απαίτηση χαμηλής ενεργειακής κατανάλωσης είναι παράγοντας που αξιολογείται σε κάθε υπολογιστικό σύστημα, από την πιο απλή φορητή συσκευή έως πολύπλοκα Ε.Σ.

Στα συστήματα που εξετάζονται, η συνολική κατανάλωσης ενέργειας ολόκληρου του συστήματος διαμορφώνεται σε συντριπτικό βαθμό από την κατανάλωση ισχύος δυο μόνο εκ των στοιχείων του, του επεξεργαστή και της μνήμης. Οι προσβάσεις στην εξωτερική μνήμη εκτός του ότι είναι αργές (όπως αναφέρθηκε προηγουμένως) έχουν και αρκετά υψηλό ενεργειακό τίμημα, κι αυτό γιατί προκειμένου να αναγνωσθεί μια εξωτερική μνήμη πρέπει να προηγηθεί πρόσβαση στην αρτηρία δεδομένων και διευθύνσεων της μνήμης. Με άλλα λόγια, πρέπει να μεταφερθούν σήματα από το εσωτερικό του ολοκληρωμένου κυκλώματος προς τα έξω και αντιστρόφως. Για να γίνει αυτό, πρέπει να χρησιμοποιηθούν ενισχυτές σημάτων, με επιπρόσθετη αύξηση της κατανάλωσης.

ΚΕΦΑΛΑΙΟ 4: ΠΑΡΑΔΕΙΓΜΑΤΑ Ε.Σ. ΣΤΗ ΒΙΟΜΗΧΑΝΙΑ

Κατόπιν ανάλυσης των βασικών αρχών και λειτουργίας των δομικών στοιχείων των Ε.Σ., σε αυτό το κεφάλαιο παρουσιάζονται ορισμένα παραδείγματα – αποτελούν ένα πολύ μικρό δείγμα των Ε.Σ. που βρίσκονται σχεδόν σε όλα τα σύγχρονα ηλεκτρικά/ηλεκτρονικά συστήματα - πρακτικών βιομηχανικών Ε.Σ. που χρησιμοποιούνται σε ποικίλους τομείς της καθημερινότητας. Στόχο του παρόντος κεφαλαίου αποτελεί, μέσα από την ανάλυση των παραδειγμάτων που ακολουθούν, μια ολοκληρωμένη επισκόπηση των ζητημάτων, όσο αυτό είναι δυνατό στα πλαίσια μιας πτυχιακής εργασίας, γύρω από τα Ε.Σ.

Στην ενότητα που ακολουθεί παρουσιάζονται ορισμένα σχεδιαστικά παραδείγματα προκειμένου να καταστεί σαφές το κομμάτι των Ε.Σ. που αφορά στον έλεγχο και τους υπολογισμούς των παραμέτρων που χρησιμοποιούνται σε συστήματα διαφόρων βαθμών πολυπλοκότητας.

4.1 Παράδειγμα πρώτο: Το Ε.Σ. μιας επαγωγικής κουζίνας [15]

Μια από τις σύγχρονες οικιακές ηλεκτρικές συσκευές στις οποίες κανείς μπορεί να διακρίνει ένα ενσωματωμένο σύστημα αποτελεί η επαγωγική εστία (Εικ.1-3), η οποία δεν έχει καμία σχέση με τις κεραμικές κουζίνες με περιστροφικούς διακόπτες του παρελθόντος, στις οποίες ασφαλώς δεν περιλαμβάνονταν ενσωματωμένα συστήματα. Στις τελευταίες, η ρύθμιση της θερμοκρασίας γίνονταν με κάποιο διμεταλλικό έλασμα, το οποίο μόλις θερμαινόταν αρκετά (διέθετε όριο

κάποιας θερμοκρασίας), κοβόταν, διακόπτοντας τη ροή του ρεύματος στην αντίσταση. Ο περιστροφικός διακόπτης ρύθμιζε την απόσταση του διμεταλλικού ελάσματος και της επαφής της αντίστασης.

Η ανάγκη για βελτίωση του σχεδιασμού της κουζίνας με σκοπό την αύξηση των δυνατοτήτων ελέγχου και λειτουργίας της αλλά και της αισθητικής ικανοποίησης του αγοραστή, οδήγησε τους κατασκευαστές στον επανασχεδιασμό του συστήματος, συμπεριλαμβάνοντας ένα Ε.Σ.

Οι αλλαγές που παρέχει το νέο Ε.Σ. είναι οι εξής:

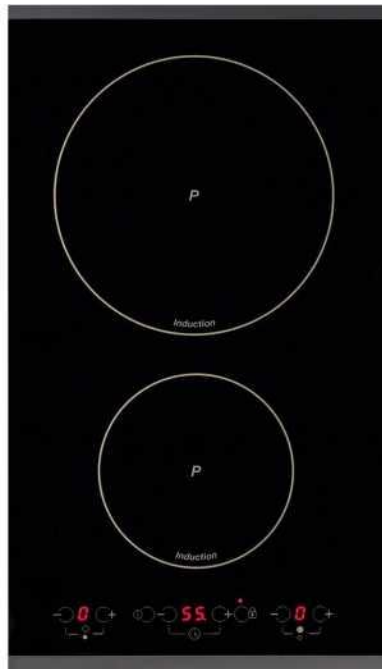
- Αντικατάσταση των περιστροφικών διακοπών με οπτικούς αισθητήρες ανάκλασης (reflective optical sensors). Τα κουμπιά δεν είναι πλέον περιστροφικά αλλά αποτελούνται από κατάλληλα αισθητήρια τοποθετημένα πίσω από τη γυάλινη εστία (λειτουργούν με υπολογισμό του χρόνου που κάνει ένα σήμα ακτινοβολίας να διανύσει μια απόσταση) και αποστέλλουν που υπεριώδη ακτινοβολία σε περίπτωση ανίχνευσης αντανάκλασης. Ο χρήστης αρκεί να τοποθετήσει το δάκτυλο πάνω από το αισθητήριο και μέσω της αντανάκλασής του ανιχνεύεται η απαραίτητη απόσταση του δακτύλου για την ενεργοποίηση της λειτουργίας θέρμανσης του «ματιού»
- επιλογή επιθυμητής ισχύος λειτουργίας της κουζίνας με αύξησης ή μείωσης θερμοκρασίας μέσω των κατάλληλων πλήκτρων (από το 0 που αντιστοιχεί σε μη αναμμένο «μάτι» έως το 9 που αντιστοιχεί στη μέγιστη θερμοκρασία) για το συγκεκριμένο «μάτι»,
- ηχητική ένδειξη (beeper) για κάθε επιλογή από τον χρήστη,
- φωτεινή ένδειξη με οθόνες επτά στοιχείων (7 segments display) για κάθε «μάτι» που υποδηλώνει τον αριθμό που έχει επιλεγεί. Ακόμη, κάθε μάτι που

επιλέγεται σηματοδοτείται από μια τελεία,

- ρύθμιση της επιθυμητής ισχύος (και κατ' επέκταση της θερμοκρασίας που αναπτύσσει το «μάτι») μέσω διαμόρφωσης εύρους παλμού (PWM, pulsewidthmodulation), όπου ανάλογα με την επιλογή της επιθυμητής ισχύος (υποδεικνύεται από το ρεύμα και την τάση λειτουργίας, που καθορίζουν την ισχύ), διαμορφώνεται καταλλήλως και η περίοδος ενεργοποίησης και απενεργοποίησης. Για παράδειγμα, θερμοκρασία επιπέδου 1 υποδεικνύει ότι η αντίσταση στο «μάτι» θα είναι ενεργοποιημένη για το 10% της περιόδου λειτουργίας και απενεργοποιημένη το υπόλοιπο 90%. Έτσι, στη μέγιστη βαθμίδα 9, η αντίσταση στο «μάτι» θα είναι ενεργοποιημένη για το 100% της περιόδου λειτουργίας,
- κλείδωμα ασφαλείας το οποίο ακυρώνει την ενεργοποίηση των διακοπών ώστε να μην πατηθούν εκ παραδρομής (π.χ. ένα αντικείμενο αφεθεί πάνω στην κουζίνα). Η ενεργοποίηση επιτυγχάνεται με παρατεταμένη επιλογή του όποιου διακόπτη για να αρθεί το κλείδωμα.

Η υλοποίηση του προγράμματος δεν περιορίζεται από στενά χρονικά πλαίσια (δηλαδή δεν απαιτείται να ενεργοποιηθεί η αντίσταση μέσα σε ορισμένους), ούτε από ενεργειακούς περιορισμούς (η εστία τροφοδοτείται από εναλλασσόμενη τάση, φασικής ενεργού τιμής τα $230V_{rms}$ από το δίκτυο και σχεδόν ολόκληρη η ηλεκτρική ισχύς καταναλώνεται με τη μορφή θερμότητας στις αντιστάσεις της συσκευής), ούτε από οικονομικής άποψης (η δαπάνη για τα υλικά κατασκευής και τον ηλεκτρικό εξοπλισμό είναι αποτελεί πολύ μικρόποσοστό της υπόλοιπης κατασκευής). Σε γενικές γραμμές, ο σχεδιασμός ενός Ε.Σ. για τη συγκεκριμένη συσκευή είναι υλοποιήσιμος σχετικά εύκολα και σε σύντομο χρονικό διάστημα, αρκεί να επιλεγεί κατάλληλος μικροελεγκτής με ικανό αριθμό εισόδων και εξόδων

ώστε να υποστηρίζονται οι επιθυμητές λειτουργίες της εφαρμογής.



Σχήμα4-1 Σύγχρονη επαγωγική εστία με ενσωματωμένο σύστημα [15]

Επειδή οι ανάγκες σε αυτήν την περίπτωση είναι λίγες, συνήθως οι κατασκευαστές επιλέγουν έναν απλό επεξεργαστή, με 8bitοργάνωση μνήμης και ρολόι στα 16MHz(σε σύγκριση με τους επεξεργαστές στους επιτραπέζιους ή φορητούς υπολογιστές οι οποίοι διαχειρίζονται μνήμες 64bitστα 1.5 με 2 GHz). Ο επεξεργαστής πρέπει να διαθέτει εσωτερική μνήμη FLASHώστε να μπορεί να διατηρεί το πρόγραμμα (ενδεικτικά, 32KB μνήμηςFLASHείναι υπεραρκετά). Η εξωτερική μνήμη δεν αποκλείεται σε αυτές τις περιπτώσεις, ωστόσο, από τη στιγμή που κοστίζουν τι ίδιο, η εσωτερική μνήμη αποτελεί τη συνηθέστερη επιλογή. Άλλωστε, η χωρητικότητα μνήμης τυχαίας προσπέλασης που απαιτείται να διαθέτει ο μικροεπεξεργαστής είναι της τάξης των μερικών KB.

Όσον αφορά τον απαιτούμενο αριθμό των διαθέσιμων ακροδεκτών, πρέπει να

ληφθούν υπόψη οι 8 έξοδοι που χρειάζονται για κάθε μια από τις οθόνες των 7 στοιχείων και της τελείας, η έξοδος για την ηχητική ειδοποίηση και η είσοδος για το σήμα του κάθε διακόπτη. Όλες αντιστάσεις διαθέτουν έναν ηλεκτρονόμο που ελέγχει τη ροή του ρεύματος. Συνεπώς, οι συνολικοί απαιτούμενοι ακροδέκτες που υπολογίζονται για λειτουργία 2 «ματιών» της κουζίνας είναι $2*8$ (για τα 7 στοιχεία και την τελεία), συν 1 για το ηχητικό σήμα, συν 2 εξόδους και 6 εισόδους, συνολικά 25 εισόδους/εξόδους (θύρες). Ωστόσο, προκειμένου να ελαττωθεί ο αριθμός των απαιτούμενων ακροδεκτών, υπάρχει η δυνατότητα χρησιμοποίησης δυο εξωτερικών σειριακών καταχωρητών (registers) παράλληλης εξόδου των 8bit, συνδεδεμένοι σε σειρά, οι οποίοι χρησιμοποιούν 3 ακροδέκτες του μικροεπεξεργαστή (και όχι $2*8=16$ ακροδέκτες). Έτσι, οι συνολικές ανάγκες του συστήματος σε ακροδέκτες μειώνονται σε $3+1+2+6 = 12$ ακροδέκτες εισόδου/εξόδου.

Εκτός από τις παραπάνω συνδέσεις, απαιτούνται και οι κατάλληλες αντιστάσεις σε ορισμένους ακροδέκτες (pull-up ή pull-down), ο σχεδιασμός της τυπωμένης πλακέτας του κυκλώματος (PCB, PrintedCircuitBoard,) σε κάποιο κατάλληλο λογισμικό (π.χ. AltiumDesigner), η δρομολόγηση των ασθενών και ισχυρών ρευμάτων μέσω καλωδίων και δρόμων χαλκού κατάλληλης διατομής, και τέλος η κατάλληλη θωράκιση, σύμφωνα με το πρότυπο της ηλεκτρομαγνητικής θωράκισης από ηλεκτρομαγνητικές παρεμβολές.

Το Ε.Σ. που περιγράφηκε παραπάνω δεν απαιτεί λειτουργικό σύστημα. Αρκεί να γραφεί ένα πρόγραμμα είτε σε γλώσσα μηχανής (assembly), είτε σε γλώσσα υψηλού επιπέδου, όπως C, C++, κ.ά. Το πρόγραμμα πρέπει να αποτελείται από έναν ατέρμων βρόχο, ο οποίος θα εκτελεί διαδοχική επισκόπηση των επιλογών των διακοπών, θα ενημερώνει κατάλληλα τις οθόνες των 7 στοιχείων, θα

ενεργοποιεί το ηχητικό σήμα και θα καθορίζει μέσω της PWM διαμόρφωσης την επιλεγμένη θερμοκρασία.

Προκειμένου να υπάρχει δυνατότητα εξυπηρέτησης των διαφορετικών αναγκών που διέπουν το καταναλωτικό κοινό, μια συσκευή (εν προκειμένω η κουζίνα) κατασκευάζεται με πολλαπλές επιλογές δυνατοτήτων. Πέραν του βασικού μοντέλου που αναλύθηκε προηγουμένως, θα μπορούσε να προστεθεί ένα δομικό στοιχείο ασύρματης επικοινωνίας τύπου Bluetooth (το οποίο απαιτεί 2 ακροδέκτες RX/TX για σειριακή επικοινωνία με τον μικροεπεξεργαστή), και να παρέχεται μια εφαρμογή για έξυπνα κινητά (smartphones), ώστε η ρύθμιση της θερμοκρασίας να γίνεται με μεγαλύτερη ευκολία και άνεση. Εναλλακτικά, θα μπορούσε να τοποθετηθεί ένα στοιχείο πρωτόκολλου επικοινωνίας wifi 802.11 για σύνδεση στο διαδίκτυο, ώστε να αποστέλλονται ενημερώσεις μέσω email ή να υπάρχει δυνατότητα ελέγχου από την κεντρική ιστοσελίδα. Τέλος, αναφέρεται ότι το κόστος κάθε μοντέλου επαγωγικής κουζίνας και εξαρτάται κυρίως από την πολυπλοκότητα του σχεδιασμού και τις παροχές που προσφέρει, που είναι σαφώς αρκετά υψηλότερη του κόστους κατασκευής.

4.2 Παράδειγμα δεύτερο: Το Ε.Σ. για έναν ψηφιακό καταγραφέα [15]

Ένα δημοφιλές Ε.Σ. αποτελεί η συσκευή καταγραφής κίνησης υψηλού βαθμού επεξεργασίας και ανάλυσης σήματος (HighDefinitionDVR), ο οποίος προορίζεται για χρήση στο μπροστινό μέρος οχήματος, με σκοπό την παρακολούθηση της πορείας που διαγράφει το όχημα. Η πληροφορία της διαδρομής που καταγράφεται μπορεί να αξιοποιηθεί σε πολλές περιπτώσεις, όπως για παράδειγμα ως αποδεικτικό στοιχείο σε περίπτωση τρακαρίσματος, ώστε να διευκρινιστούν οι

συνθήκες του ατυχήματος. Αν συγκριθεί με το Ε.Σ. που παρουσιάστηκε νωρίτερα, το συγκεκριμένο έχει υψηλότερες απαιτήσεις σχεδιασμού και λειτουργίας του συστήματος, οι οποίες αναφέρονται παρακάτω:

- Αποθήκευση VIDEOυψηλής ανάλυσης,
- εργαλεία συμπίεσης του καταγραφέντος αρχείου ήχου και εικόνας προκειμένου να καταλαμβάνει λιγότερη μνήμη του συστήματος,
- χαμηλή ενεργειακή κατανάλωση, στο επίπεδο που καταναλώνει η φόρτιση μιας μικρής φορητής συσκευής,
- σημαντικό αριθμό λειτουργιών και μέσων ελέγχου των παραμέτρων του συστήματος,
- λειτουργία αναφοράς ώρας και ημερομηνίας,
- συσσωρευτή τεχνολογίας λιθίου πολυμερούς (LiPO)με συσκευή φόρτισης μέσω miniUSB,
- πλήθος επιλογών ρύθμισης της ποιότητας ανάλυσης ανάλογα με τις εκάστοτε απαιτήσεις, καθώς και διαχείρισης της κάρτας μνήμης SD,
- μετατροπέας αναλογικού σήματος σε ψηφιακό, για την ψηφιακή επεξεργασία του σήματος και τη διασύνδεση με τον οπτικό αισθητήρα (CMOSή CCD),
- υποστήριξη οθόνηςLCD (liquid crystal display),
- υποστήριξη συστήματος για τη δημιουργία συμβατότητας (μέσω δημιουργίας αρχείουFATστην κάρτα SD), για μεταφορά των δεδομένων σε υπολογιστή,
- υποστήριξη για USBMassStorageDeviceόταν συνδέεται σε έναν υπολογιστή,

- επαρκώς μικρό μέγεθος ώστε να μην διακόπτει την ορατότητα του οδηγού του οχήματος στο οποίο τοποθετείται.



Σχήμα 4-2 Ψηφιακός καταγραφέας υψηλής ανάλυσης [15]

Όπως προκύπτει από τις παραπάνω απαιτήσεις, το εν λόγω Ε.Σ. σύστημα είναι περισσότερο πολύπλοκο από το πρώτο Ε.Σ. για το οποίο έγινε λόγος προτύτερα, με πιο κρίσιμη απαίτηση αυτήν για υψηλής ποιότητας (ανάλυσης) επεξεργασίας ψηφιακού σήματος, χαρακτηριστικό που απαιτεί μνήμη αρκετά μεγάλου μεγέθους και ρυθμό επεξεργασίας μερικών δεκάδων εικονοστοιχείων το δευτερόλεπτο(FPS,framespersecond),για τη συμπίεση και την αποθήκευση σε ένα μέσο αποθήκευσης. Αυτό σημαίνει ότι ο επεξεργαστής θα πρέπει να διαθέτει οργάνωση μνήμης των32bit(με μνήμη των 8bitθα υπήρχε μεγάλη καθυστέρηση). Επιπλέον, θα πρέπει να επιτυγχάνεται χαμηλή κατανάλωση ισχύος και μικρός όγκος, αλλά και να ενσωματώνονται οι περισσότερες λειτουργίες πάνω στο ίδιο ολοκληρωμένο κύκλωμα. Γι' αυτόν τον λόγο, θα πρέπει να χρησιμοποιηθεί είτε ένας δυνατός 32biteπεξεργαστής. Θα πρέπει το ίδιο το ολοκληρωμένο κύκλωμα να υποστηρίζει

θύρα USBγια τη σύνδεση με υπολογιστή και να υποστηρίζει ανάγνωση της κάρτας μνήμης SD.Επιπλέον, για την υποστήριξη της λειτουργίας ρολογιού πραγματικού χρόνου(RTC, realtimeclock,) στο ίδιο το ολοκληρωμένο, ο κατάλληλος ακροδέκτης του επεξεργαστή που είναι υπεύθυνος για τη διατήρηση της τρέχουσας ώρας και ημερομηνίας πρέπει να ενημερώνεται μέσω κατάλληλης τροφοδοσίας από συσσωρευτή που διαθέτει η συσκευή. Επίσης, θα υπάρχει η δυνατότητα ρύθμισης του RTCσε μια συγκεκριμένη ώρα. Τοολοκληρωμένο θα πρέπει να έχει και τον ελεγκτή για τηνLCD οθόνη, ενώ τα πλήκτρα οδηγούνται άμεσα στο ίδιο το ολοκληρωμένο, είτε σε ξεχωριστούς περιφερειακούς ακροδέκτες I/O, οι οποίοι συνδέονται με τον επεξεργαστή. Επιπρόσθετα, τα κυκλώματα της φόρτισης αλλά και της προστασίας του συσσωρευτή του συστήματος επιτυγχάνεται με κατάλληλες διατάξεις, οι οποίες αποτελούν ηλεκτρικά κυκλώματα με συστοιχίες πυκνωτών και αντιστάσεων, σε κατάλληλη διάταξη. Τα στοιχεία αυτά χρησιμοποιούνται και για τη διασύνδεση των περιφερειακών ακροδεκτών.

Όσον αφορά το λογισμικό του Ε.Σ., ο προγραμματισμός του μπορεί να υλοποιηθεί είτε αποκλειστικά σε γλώσσα υψηλού επιπέδου, άνευ λειτουργικού συστήματος, κάτι αρκετά πολύπλοκο για τον σχεδιαστή, αφού θα πρέπει να υλοποιήσει το λογισμικό ελέγχου και οδήγησης των συσκευών καταγραφής, να σχεδιάσει τη διεπαφή χρήστη και να αναπτύξει το λογισμικό που αφορά στη διασύνδεση με τον υπολογιστή. Η υλοποίηση τούτη ονομάζεται «baremetalprogramming» και είναι αρκετά απαιτητική, αλλά οδηγεί στην αποδοτικότερη χρησιμοποίηση των διαθέσιμων πόρων. Εναλλακτικά, θα μπορούσε να χρησιμοποιηθεί ένα λειτουργικό σύστημα (Λ.Σ.), π.χ. Linux, το οποίο διαθέτει έτοιμες βιβλιοθήκες, κατάλληλες για πολλές λειτουργίες. Το μειονέκτημα χρήσης Λ.Σ. είναι ότι απαιτείται πιο γρήγορος επεξεργαστής και μεγαλύτερη μνήμη, αφού το ίδιο το Λ.Σ. χρησιμοποιεί κάποιους από τους πόρους του συστήματος. Στον αντίποδα, η

ανάπτυξη του λογισμικού όταν υπάρχει ένα Λ.Σ. γίνεται πολύ πιο εύκολη, αφού το Λ.Σ. παρέχει στον προγραμματιστή ένα φιλικό περιβάλλον προγραμματισμού, πιο κοντά στο επίπεδο σκέψης του, παρά στο επίπεδο του υλικού (hardware).

4.3 Παράδειγμα Τρίτο: Το Ε.Σ. για ένα αυτοκίνητο όχημα[15]

Στην περίπτωση του σύγχρονου αυτοκινήτου δεν υπάρχει μόνο ένα Ε.Σ., αλλά πλήθος διαφορετικών Ε.Σ., που συνεργάζονται μεταξύ τους για την υλοποίηση τελικού προϊόντος, ώστε να πληροί τις απαιτήσεις ασφάλειας και λειτουργικότητας των επιβαινόντων. Εφόσον όλα αυτά τα συστήματα συνεργάζονται μεταξύ τους, μπορούν να θεωρηθούν ως ένα μεγάλο Ε.Σ. Η τοποθέτηση ψηφιακών συστημάτων μέσα στο αυτοκίνητο ρυθμίζει κάθε στάδιο λειτουργίας του κινητήρα και συμβάλει στην βελτιστοποίηση της καύσης. Οι λειτουργίες του Ε.Σ. σε ένα αυτοκίνητο είναι οι:

- βελτιστοποίηση της καύσης,
- έλεγχος οδήγησης (ACC, Adaptive Cruise Control),
- προειδοποίηση σύγκρουσης (FCW, Forward Collision Warning),
- προειδοποίηση αποχώρησης λωρίδας (LCW, Lane Departure Warning),
- σύστημα βοήθειας στο παρκάρισμα (PAM, Park Assist System),
- σύστημα παρακολούθησης πίεσης ελαστικών (TPMS, Tire Pressure Monitoring System)
- ξεκλείδωμα χωρίς κλειδί (RemoteKeylessEntry),
- επικοινωνία με συνήθη πρωτόκολλα διασύνδεσης, όπως bluetooth, wifi, GSM,
- λειτουργία πλοήγησης (GPS),

- διαχείριση λειτουργίας του ραδιοφώνου (π.χ. όταν λαμβάνεται μια κλήση μέσω bluetooth μειώνεται η ένταση),
- σύστημα αντιμπλοκαρίσματος τροχών (ABS, Antilock braking system),
- έλεγχος ταχύτητας (Wheel Speed Sensor),
- κάμερα οπισθοπορείας (rear view camera),
- ηλεκτρονικό τιμόνι (electric power steering),
- έλεγχος αερόσακων (airbags),
- έλεγχος ζώνης (seatbelt pretensioner),
- μπροστινό Radar για άμεση ενεργοποίηση φρένων,
- διαγνωστικά αισθητηρίων (sensor diagnostics),
- Event Data Recorders, κ.ά.

Ο αριθμός των Ε.Σ. στα αυτοκίνητα έχει αυξηθεί σημαντικά τις τελευταίες δύο δεκαετίες. Μια ηλεκτρονική αρχιτεκτονική βασισμένη σε κυκλώματα μέσα σε ένα αυτοκίνητο (in-vehicle) είναι μια πολύπλοκη διαδικασία, η ανάπτυξη της οποίας συμπεριλαμβάνει διάφορους κατασκευαστές και προμηθευτές. Βασικές απαιτήσεις στην αναπτυξιακή διαδικασία είναι οι απαιτήσεις ασφαλείας, η αξιολόγηση σε πραγματικό χρόνο, ο σχεδιασμός, κ.ά.

Στη βιομηχανία των αυτοκινήτων χρησιμοποιούνται κατά κόρον δύο οικογένειες ενσωματωμένων συστημάτων: τα DSP (Digital Signal Processors) που είναι πλατφόρμα καθαρού λογισμικού, με πυρήνα έναν μικροεπεξεργαστή και κάποια περιφερειακά και τα FPGA (Field Programmable Gate Array ή συστοιχία επιτόπια προγραμματιζόμενων πυλών), που είναι κυρίως υλικό. Τα FPGAs παρουσιάζουν σημαντικό πλεονέκτημα σε εφαρμογές που απαιτούν πολύ μεγάλες ταχύτητες.

Στη συνέχεια, υπάρχει το CAN (Controller Area Network) το οποίο είναι το πρωτόκολλο επικοινωνίας μεταξύ του οποίου επικοινωνούν όλοι οι μικροελεγκτές σε ένα αυτοκίνητο. Τα καλώδια επικοινωνίας του CAN είναι συνεστραμμένα καλώδια, τα οποία είναι ανθεκτικά στην ηλεκτρομαγνητική ακτινοβολία. Οι εφαρμογές του CAN στα αυτοκίνητα περιλαμβάνουν τον έλεγχο της μηχανής, τις επικοινωνίες, τον έλεγχο του αμαξώματος και τις πληροφορίες ταξιδιού. Συνήθως τα αυτοκίνητα έχουν ένα ή δυο δίκτυα CAN.

Το ένα CAN περιλαμβάνει τα αισθητήρια και τις συσκευές η λειτουργία των οποίων βρίσκεται σε προτεραιότητα στην ιεραρχία εκτέλεσης διεργασιών από τον μικροεπεξεργαστή, προκειμένου να μπορούν να υλοποιούνται άμεσα χειρισμοί που σχετίζονται με το σύστημα ABS, τον έλεγχο της πέδησης, τον έλεγχο του τιμονιού, τον έλεγχο των φώτων, τον έλεγχο του κινητήρα κ.ά.

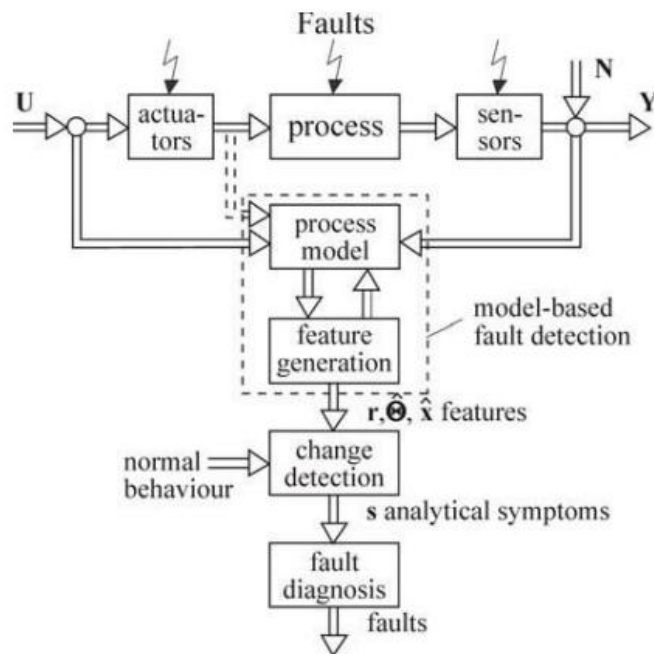
Στο άλλο δίκτυο CAN τοποθετούνται τα αισθητήρια και οι συσκευές που κατέχουν μικρότερη προτεραιότητα στην ιεραρχία εκτέλεσης διεργασιών από τον μικροεπεξεργαστή, καθώς η εκτέλεσή τους είναι επικουρική και όχι πρωταρχικής σημασίας για το σύστημα. Τέτοιες λειτουργίες επιτελούν διάφοροι αισθητήρες όπως οι λυχνίες ένδειξης της κατάστασης των θυρών του οχήματος (ανοιχτές/κλειστές), της χρήσης ζώνης από τους επιβαίνοντες, του κλιματισμού και της θερμοκρασίας κ.ά. Η εξέλιξη της τεχνολογίας επιβάλλει αύξηση των παραπάνω επικουρικών λειτουργιών του συστήματος με σύνδεση ακόμα περισσότερων αισθητήρων στο CAN, προκειμένου να καταστεί το όχημα ακόμα πιο ασφαλές αλλά και εύχρηστο τόσο για τον οδηγό όσο και για τους επιβάτες. Στη βάση αυτή, τεχνικές άμεσης τοποθέτησης των αισθητήρων με ενδεικτικές λυχνίες στο ταμπλό του οχήματος που εφαρμόζονταν στα συστήματα παλαιότερης τεχνολογίας έχουν αντικατασταθεί από σύγχρονα, τα οποία ενημερώνουν τον

αντίστοιχο επεξεργαστή, δηλαδή τη μονάδα ελέγχου μηχανής (ECU, EngineControl Unit), και με αυτόν τον τρόπο εκτελούνται οι κατάλληλες εντολές/ενέργειες από αυτήν. Στην περίπτωση αυτού του Ε.Σ. απαιτούνται τα εξής: ένα Λ.Σ. που να ανταπεξέρχεται στα αυστηρά χρονικά περιθώρια απόκρισης του συστήματος, επεξεργαστές με ταχύτατη εξυπηρέτηση των διακοπών και μονάδα διαχείρισης μνήμης ή μονάδα προστασίας μνήμης (MPU), -αφού απαιτούνται αρκετές διεργασίες που πρέπει να είναι απομονωμένες, ώστε η μια να μην καταλαμβάνει τον χώρο μνήμης της άλλης-, αρκετά πολύπλοκες συνδέσεις των αισθητήρων, ένα Λ.Σ. για τη διαχείριση των επικουρικών δικτύων CAN όπου θα σχεδιαστούν οι γραφικές διεπαφές και, τέλος, τηλεματική -αποστολή και λήψη πληροφοριών μέσω ασύρματου δικτύου δεδομένων-, ώστε σε περίπτωση ατυχήματος ή βλάβης να παρέχεται η δυνατότητα αποστολής της τοποθεσίας για τον εντοπισμό του οχήματος. Παράλληλα, κάθε ηλεκτρονικό κύκλωμα/στοιχείο του οχήματος οφείλει να υποστηρίζει λειτουργία ενημέρωσης σε περίπτωση βλάβης, ώστε να διακόπτεται η λειτουργία του και να μεταβαίνει σε κατάσταση ασφαλούς λειτουργίας ή αδράνειας, ώστε να μην επηρεάζει τις λειτουργίες του οχήματος.

Στα σύγχρονα οχήματα έχει καθιερωθεί η ύπαρξη κατάλληλου σήματος με φωτεινή ένδειξη δυσλειτουργίας (MIL, Malfunction Indicator Lamp) ώστε να ειδοποιείται ο οδηγός του οχήματος για την ύπαρξη προβληματικής κατάστασης σε κάποιο τμήμα του οχήματος. Για την επίτευξη του παραπάνω συστήματος ασφαλείας, τα οχήματα διαθέτουν περισσότερα του ενός δίκτυα επικοινωνίας με τους αισθητήρες, ώστε να εξασφαλίζεται η ενημέρωση ανίχνευσης βλάβης, ακόμα και στην περίπτωση βλάβης του ίδιου του καναλιού που είναι υπεύθυνο για την ενημέρωση. Το Ε.Σ. απαιτεί επίσης μνήμη Flash ή EEPROM, για να αποθηκεύει σε μη πτητικό μέσο το ενημερωμένο για το ιστορικό βλαβών αρχείο (fault log file),

ώστε να ελέγχονται ανά πάσα στιγμή οι συνθήκες εμφάνισης του εκάστοτε σφάλματος, κάτι που μπορεί να φανεί χρήσιμο στον τακτικό έλεγχο του οχήματος.

Τα αισθητήρια του CAN η λειτουργία των οποίων βρίσκεται σε προτεραιότητα στην ιεραρχία εκτέλεσης διεργασιών αναγκαία συνδέονται σε ανεξάρτητη γραμμή διακοπής (interrupt line), ώστε να το σύστημα να ανταποκρίνεται άμεσα στις εντολές που υπαγορεύονται από αυτά, σε αντίθεση με τα αισθητήρια του CAN που βρίσκονται σε χαμηλή προτεραιότητα στην ιεραρχία εκτέλεσης των διεργασιών του συστήματος και δεν απαιτούν ανεξάρτητη γραμμή επικοινωνίας, αλλά εντάσσονται στις διεργασίες που εκτελούνται από το Λ.Σ. Ασφαλώς, όπως αναφέρθηκε και πρωτύτερα, πρέπει να εξεταστεί ο μεγαλύτερος χρόνος χειρισμού βλάβης (worst case service time), με στόχο την όσο το δυνατό ελαχιστοποίησή του, ώστε να αυξάνεται ο βαθμός της αξιοπιστίας του συστήματος ασφάλειας. Σημαντικό πλεονέκτημα των συστημάτων ασφαλείας που περιγράφηκαν ανωτέρω είναι ότι η ανάπτυξη του κώδικα που απαιτείται για τον προγραμματισμό του υλικού γίνεται σε γλώσσα υψηλού επιπέδου, γεγονός που διευκολύνει τους προγραμματιστές. Εξαιρέση αποτελούν οι λειτουργίες για τις οποίες απαιτείται πολύ συγκεκριμένος χρόνος και σειρά υλοποίησης, και ο καλύτερος τρόπος να επιτευχθεί η υψηλή απόδοση της απόκρισης του επεξεργαστή όσο αφορά αυτές είναι με προγραμματισμό στη γλώσσα μηχανής του.

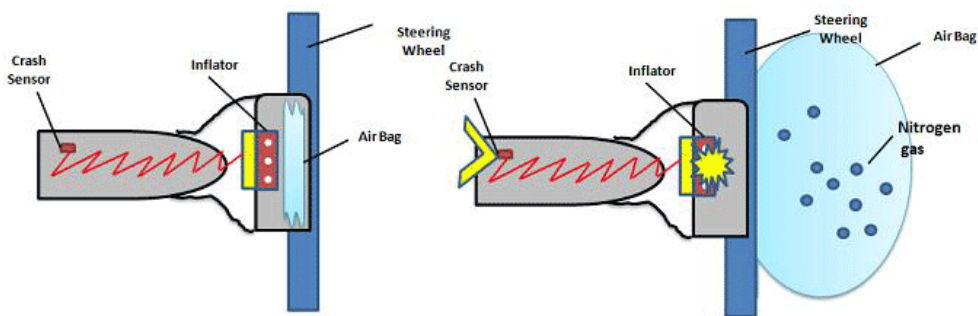


Σχήμα 4-3 Σύστημα διάγνωσης λαθών βασισμένο σε μοντέλο όλου του συστήματος αυτοκινήτου[16]

4.3.1 Το σύστημα του αερόσακου [15]

Οι αερόσακοι είναι παθητικοί μηχανισμοί ασφαλείας και είναι υποχρεωτικοί σε όλα τα αυτοκίνητα. Σκοπός του αερόσακου, ο οποίος ενεργοποιείται όταν υπάρξει κάποιο τρακάρισμα, είναι να αποτρέξει την σύγκρουση των επιβατών με το εσωτερικό του οχήματος. Οι αερόσακοι συνήθως είναι φτιαγμένοι από νάιλον και είναι κρυμμένοι πίσω από διάφορα πάνελ στο εσωτερικό του αυτοκινήτου, όπως π.χ. το τιμόνι. Ανάλογα με την σφοδρότητα της πρόσκρουσης, ο αισθητήρας πρόσκρουσης (crash sensor) στέλνει σήμα στη μονάδα ελέγχου του αερόσακου. Αυτή η μονάδα ενεργοποιεί το μηχανισμό ανάφλεξης ο οποίος παράγει αέριο αζώτο. Ο χρόνος μεταξύ της ανίχνευσης πρόσκρουσης και της πλήρους

ανάπτυξης του αερόσακου είναι μερικά εκατοστά του δευτερολέπτου και η ταχύτητα του αερόσακου είναι μερικές εκατοντάδες μίλια την ώρα, γεγονός που μπορεί να καταστεί επιβλαβές για τον επιβάτη.



Σχήμα 4-4 Σύστημα αερόσακου [15]

Ένα σύστημα αερόσακων περιλαμβάνει πλήθος αισθητήριων συστημάτων. Η λειτουργία των ποικίλων αισθητηρίων που διατίθενται στο εμπόριο και προορίζονται για το εν λόγω σύστημα στηρίζεται στη μέτρηση της επιτάχυνσης που αναπτύσσεται κατά τη σύγκρουση με αισθητήρες επιτάχυνσης που υπάρχουν πάνω και περιφερειακά του οχήματος. Επιπλέον, εντός της Κ.Η.Μ.Ε.(Κεντρική Ηλεκτρονική Μονάδα Επεξεργασίας, συχνά αποκαλούμενη ως «Εγκέφαλος») υπάρχει ενσωματωμένος ένας αισθητήρας επιτάχυνσης (κεντρικός).

Όσον αφορά τους αερόσακους, υπάρχουν συστήματα με γεννήτρια αερίων διπλής φάσης, στα οποία, ανάλογα με τον κατασκευαστή, περιλαμβάνονται οι παρακάτω αισθητήρες:

- Ένας αισθητήρας ανατροπής (angular rate sensor), ο οποίος βρίσκεται ενσωματωμένος εντός της Κ.Η.Μ.Ε. και είναι υπεύθυνος για τον υπολογισμό του ρυθμού περιστροφής του οχήματος γύρω από τον διαμήκη άξονα,
- συστήματα ανίχνευσης πρόσδεσης των επιβατών στις ζώνες ασφαλείας του οχήματος,
- αισθητήρας ρύθμισης της απόστασης του καθίσματος από το τιμόνι, για περισσότερη άνεση στην οδήγηση,
- σύστημα ανίχνευσης μάζας στα καθίσματα με χρήση αισθητήρων πίεσης, τα οποία ανιχνεύουν επαρκές βάρος ώστε να αποφανθούν για την ύπαρξη επιβάτη στο κάθισμα και
- μία κάμερα τοποθετημένη σε σημείο κοντά στον εσωτερικό καθρέπτη, για την ανίχνευση παρουσίας των επιβαινόντων στην περιοχή εμβέλειας του αερόσακου.

Για την περίπτωση των παράπλευρων συγκρούσεων, διατίθενται αισθητήρες πίεσης παραπλεύρως του οχήματος. Η Κ.Η.Μ.Ε. λαμβάνει διαρκώς τα σήματα των διάφορων αισθητήρων -συνήθως πρόκειται για αισθητήρες επιτάχυνσης ή πίεσης- και εκτελεί συγκρίσεις με καταχωρημένες τιμές σε αρχεία της μη πτητικής μνήμης του συστήματος, ώστε να αποφανθεί για το αν πρόκειται για φυσιολογικές συνθήκες του οχήματος ή αν οι τιμές ξεπερνούν τα επιτρεπόμενα όρια. Στην τελευταία περίπτωση, η Κ.Η.Μ.Ε. αποστέλλει εντολή ενεργοποίησης των μονάδων των αερόσακων σε πολύ σύντομο χρονικό διάστημα, της τάξης των μερικών ms (microseconds). Επιπλέον, μια ζωτικής σημασίας λειτουργία της Κ.Η.Μ.Ε. είναι η αξιολόγηση της κατάστασης των αισθητήρων, δηλαδή αν οι αισθητήρες λειτουργούν ομαλά ή παρουσιάζουν

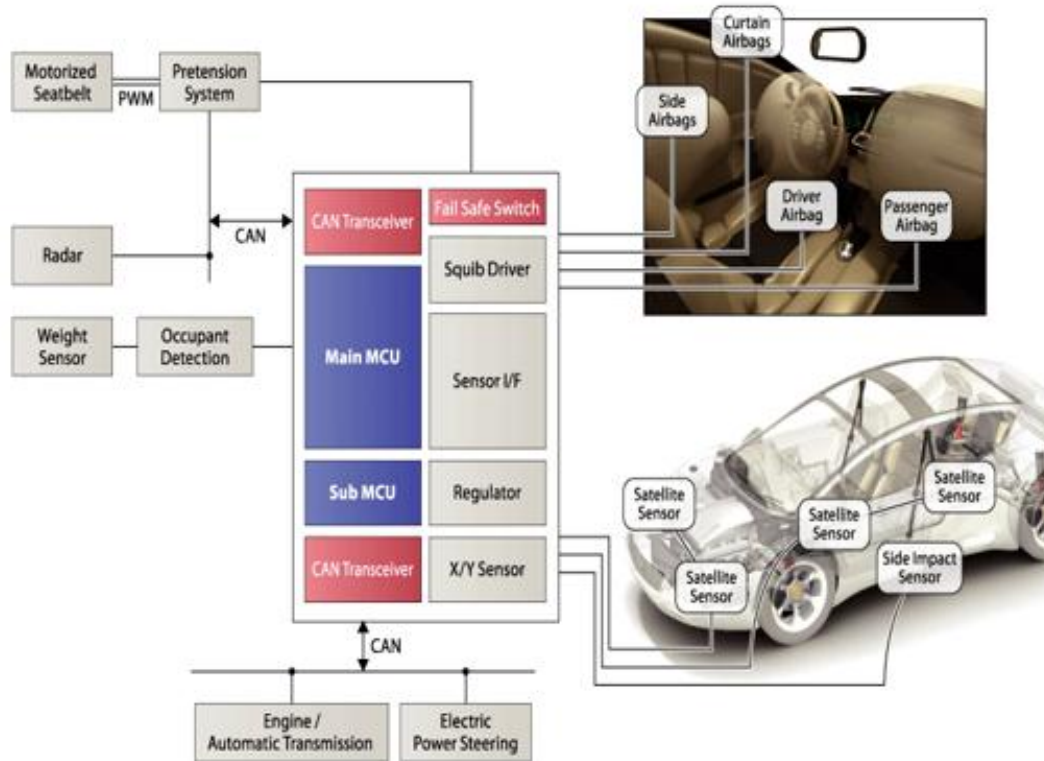
σφάλματα, ώστε να θέτει το σύστημα αερόσακων εκτός λειτουργίας και να παρέχει σήμα ενημέρωσης δυσλειτουργίας.

Προκειμένου να επιτευχθεί η αξιόπιστη λειτουργία της Κ.Η.Μ.Ε. υπό οποιεσδήποτε συνθήκες, αυτή οφείλει να τοποθετείται σε τέτοιο μέρος στο εσωτερικό του οχήματος ώστε να εξασφαλίζεται η προστασία της ακόμα και κατόπιν σύγκρουσης. Για συστήματα παράπλευρων αερόσακων, υπάρχουν δυο λογικές διαχείρισής τους, αυτή που τους διαχειρίζεται κεντρικά, ως επιπρόσθετους των εμπρόσθιων αερόσακων, κι εκείνη που διαθέτει ανεξάρτητη μονάδα ελέγχου για αυτούς.



Σχήμα 4-5 Κεντρική Ηλεκτρονική Μονάδα Επεξεργασίας [15]

Στο παρακάτω σχεδιάγραμμα παρουσιάζεται ο τρόπος με τον οποίο το CAN επικοινωνεί με την MCU και τους διάφορους αισθητήρες:



Σχήμα 4-6 Επικοινωνία του CAN με την MCU [15]

4.3.2 Event Data Recorders (EDRs)

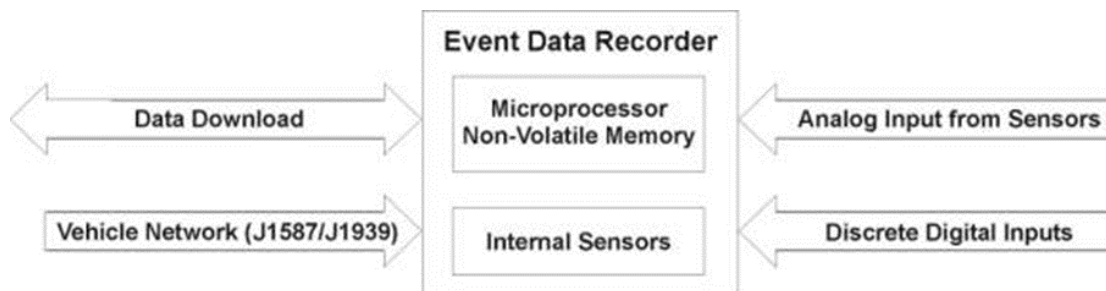
Το Event Data Recorders είναι το σύστημα που συνεχώς καταγράφει πληροφορίες που σχετίζονται με την κατάσταση του αυτοκινήτου. Σε περίπτωση ατυχήματος ο καταγραφέας σώζει τις πληροφορίες που είχε καταγράψει μερικά δευτερόλεπτα ακριβώς πριν και ίσως μετά την σύγκρουση, εάν έχει μείνει ανέπαφος. Τα EDRs μπορεί να είναι ανεξάρτητες ηλεκτρονικές μονάδες ελέγχου ή μπορεί να είναι

μέρος άλλων ηλεκτρικών μονάδων όπως της Κεντρικής Ηλεκτρονικής Μονάδας Επεξεργασίας η του κυκλώματος του αερόσακου.

Σε αντίθεση με τους καταγραφείς ατυχημάτων (accident recorders), τα οποία είναι συστήματα που μπαίνουν επιπρόσθετα στο αυτοκίνητο και συνήθως καταγράφουν βίντεο και το στίγμα της τοποθεσίας μέσω GPS, τα EDRs είναι εγκατεστημένα στο όχημα από τον κατασκευαστή και αλληλοεπιδρούν με τα υπόλοιπα συστήματα και τους αισθητήρες. Τα περισσότερα EDRs καταγράφουν διάφορες παραμέτρους του οχήματος όπως την ταχύτητά του, τη θέση των πεταλιών, τη θέση που βρίσκεται το τιμόνι και άλλες πληροφορίες που μπορεί να είναι σχετικές με την εξακρίβωση των συνθηκών κάτω από τις οποίες έγινε ένα ατύχημα. Σύμφωνα με επίσημα πρότυπα (standards), κάθε αυτοκίνητο πρέπει να είναι εξοπλισμένο με EDRs, τα οποία πρέπει να καταγράφουν τις παρακάτω πληροφορίες :

- την αλλαγή της ταχύτητάς του οχήματος σε περίπτωση σύγκρουσης,
- την ώρα που συμβαίνει η μέγιστη αλλαγή ταχύτητας σε περίπτωση σύγκρουσης,
- το ποσοστό κλίσης της πεταλούδας γκαζιού (πόσο πιεσμένο ήταν το πετάλι επιτάχυνσης),
- την ταχύτητα του οχήματος,
- εάν είχε πατηθεί το πεντάλ του φρένου,
- πόσοι κύκλοι μηχανής εφαρμόστηκαν στο EDR κατά τη διάρκεια του ατυχήματος,
- εάν ο οδηγός χρησιμοποιούσε ζώνη ασφαλείας,
- εάν ήταν αναμμένο το λαμπάκι ένδειξης του μπροστινού αερόσακου,
- την ώρα που άνοιξε ο αερόσακος του οδηγού,

- την ώρα που άνοιξε ο αερόσακος του συνοδηγού,
- την κλίση του τιμονιού,
- την κλίση του αυτοκινήτου σε περίπτωση αναποδογυρίσματος,
- τη θέση και την κλίση των καθισμάτων,
- το χρονικό διάστημα μεταξύ των δύο πρώτων συμβάντων του ατυχήματος και
- εάν το EDR ολοκλήρωσε την καταγραφή του ατυχήματος (σε περίπτωση καταστροφής του EDR).



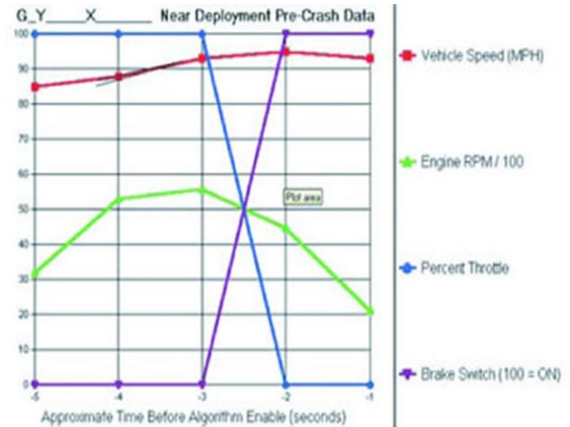
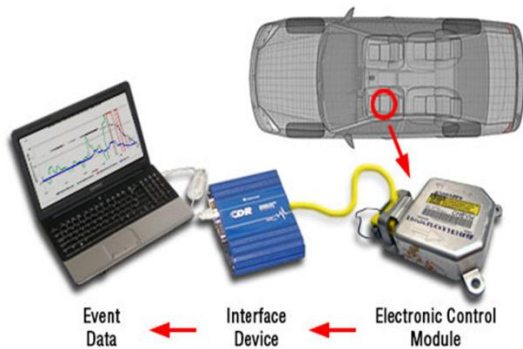
Σχήμα 4-7 Τοπικός EDA [15]

Τα επιμέρους τμήματα ενός τυπικού EDA είναι τα ακόλουθα:

- Εσωτερικοί αισθητήρες (Internal Sensors). Οι εσωτερικοί αισθητήρες βρίσκονται μέσα σε ένα EDR, όπου τα δεδομένα δεν είναι διαθέσιμα να χρησιμοποιηθούν κατευθείαν από άλλα κυκλώματα στο όχημα.
- Αναλογικό σήμα από τους αισθητήρες (Analog input from Sensors). Το αναλογικό σήμα από τους αισθητήρες είναι ένα σήμα συνεχούς τάσης της τάξεως των μερικών Volt (1 έως 5 VDC), το οποίο λαμβάνεται από τους εκάστοτε αισθητήρες που είναι τοποθετημένοι σε ένα αυτοκίνητο. Αυτό το

αναλογικό σήμα δεν εισέρχεται κατευθείαν στο EDR, αλλά ενισχυμένο, επειδή το σήμα πιθανώς να έχει εξασθενήσει λόγω του μεγάλου μήκους των καλωδίων.

- Διακριτά Σήματα Εισόδων (Discrete Digital inputs). Αυτά τα σήματα είναι οι συνδέσεις του οχήματος με on/off συσκευές. Τα φώτα των φρένων, τα alarm, η κόρνα, τα φώτα πορείας και οι προβολείς είναι on/off συσκευές που παράγουν τέτοια σήματα. Η ανάλυση της στάθμης αυτών των διακριτών σημάτων είναι ακριβή και χρονοβόρα και δεν επηρεάζει τον χειρισμό της συσκευής.
- Δίκτυο του Οχήματος (Vehicle Network). Άλλη μια δαπανηρή μέθοδος είναι συλλογή των δεδομένων από το δίκτυο του οχήματος. Τα δίκτυα σε ένα μεγάλο όχημα συνήθως είναι δύο, το χαμηλής ταχύτητας δίκτυο (SAE J1708/J1587) και το υψηλής ταχύτητας δίκτυο (SAE J1939). Όταν υπάρχουν και τα δύο δίκτυα, το χαμηλής ταχύτητας διαχειρίζεται τα γενικά δεδομένα της λειτουργίας του οχήματος και το υψηλής ταχύτητας αυτά του ελέγχου λειτουργίας της μηχανής.
- Ανάκτηση των δεδομένων (Data Download). Η διαδικασία ανάκτησης των δεδομένων είτε προς είτε από το EDR είναι μια δικατευθυντήρια σύνδεση. Για να γίνει η ανάκτηση των δεδομένων το EDR δέχεται εντολές από κάποια άλλη συσκευή π.χ. έναν υπολογιστή, και του μεταφέρει τα δεδομένα. Πληροφορίες ελέγχου (Checksums) μεταφέρονται στο τέλος κάθε μεταφοράς για την αποφυγή λαθών κατά τη μετάδοση. Επιπρόσθετα, με την ίδια διαδικασία μπορεί να κάνουμε γίνει αναβάθμιση (update) του λογισμικού που χρησιμοποιεί το EDR. Αυτή η διασύνδεση συνήθως γίνεται ενσύρματα μέσω σειριακής θύρας RS-232.



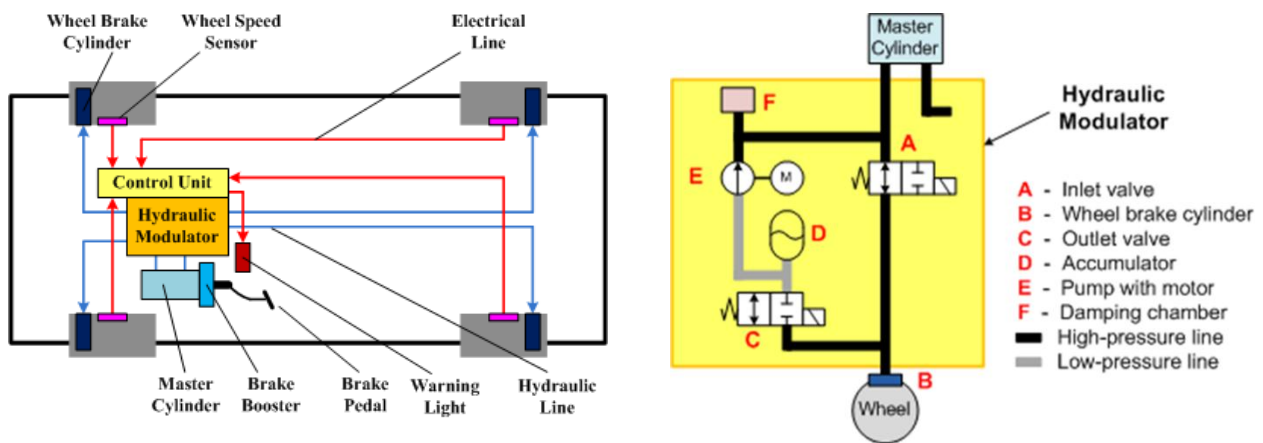
Σχήμα 4-8 Ανάκτηση δεδομένων από ένα EDR μέσω ενός υπολογιστή [15]

4.3.3 Σύστημα Αντιμπλοκαρίσματος Τροχών ABS (Anti-lock Brake System)

Το ABS είναι ένα σύστημα που αποτρέπει το μπλοκάρισμα των τροχών προσαρμόζοντας την πίεση των φρένων. Το συγκεκριμένο σύστημα παίζει έναν από τους βασικότερους ρόλους στην βελτίωση της ασφάλειας των μοντέρνων οχημάτων. Σε ολισθηρούς δρόμους και σε λείες επιφάνειες, ο χειριστής του οχήματος μπορεί να πατήσει το φρένο με μεγάλη πίεση, με αποτέλεσμα ένας ή περισσότεροι τροχοί να κλειδώσουν και το όχημα να ολισθαίνει πάνω στην επιφάνεια. Αυτό μπορεί να έχει ως αποτέλεσμα είτε το φρενάρισμα να διαρκέσει περισσότερο μέχρι να ακινητοποιηθεί το όχημα είτε ο οδηγός να μην έχει την δυνατότητα να στρίψει το όχημα είτε την αστάθεια του οχήματος.

Το σύστημα ABS ελέγχει την ταχύτητα του τροχού σε πραγματικό χρόνο και αυξομειώνει την πίεση στα φρένα αυτόματα, ώστε να μην υπάρξει μπλοκάρισμα του τροχού και ο οδηγός να έχει τον έλεγχο του οχήματος καθ' όλη τη διάρκεια.

Πλέον, το ABS συνδυάζεται και με άλλα συστήματα, όπως το ESC (Electronic Stability Control) και το Traction Control για περαιτέρω ασφάλεια του οδηγού. Τα βασικά μέρη αυτών των συστημάτων είναι τα φρένα (το πεντάλ του φρένου, οι υδραυλικοί κύλινδροι και οι ηλεκτρικές γραμμές), οι αισθητήρες ταχύτητας τροχού και ένας υδραυλικός διαμορφωτής που τον διαχειρίζεται μια ηλεκτρονική μονάδα ελέγχου.



Σχήμα 4.9 Η αρχιτεκτονική του συστήματος ABS [15]

Σε κανονικές συνθήκες η βαλβίδα εξαγωγής αέρα (C) του υδραυλικού συντονιστή είναι κλειστή και η βαλβίδα εισαγωγής αέρα (A) μένει ανοικτή μέχρι η πίεση να φτάσει στα επιθυμητά επίπεδα. Έπειτα τόσο η βαλβίδα εισαγωγής όσο και η εξαγωγής αέρα παραμένουν κλειστές για να διατηρήσουν την πίεση και να παρέχουν ικανοποιητική ροπή πέδησης για τους κυλίνδρους φρεναρίσματος. Μόλις η μονάδα ελέγχου ανιχνεύσει κάποια εξέχουσα διαφυγή του τροχού, η βαλβίδα εξαγωγής αέρα ανοίγει για να ελευθερώσει την πίεση στον συσσωρευτή (D) και να αποτρέψει πιθανό μπλοκάρισμα των τροχών. Το επιπλέον υγρό φρένων επιστρέφεται στο κύριο κύλινδρο μέσω της αντλίας του κινητήρα (E). Όταν πλέον δεν υπάρχει διαφυγή του τροχού οι ηλεκτρομαγνητικές βαλβίδες εισαγωγής και εξαγωγής αέρα αποφορτίζονται και ο υδραυλικός συντονιστής συνεχίζει την κανονική διαδικασία φρεναρίσματος.

ΚΕΦΑΛΑΙΟ 5: ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα πτυχιακή εργασία παρουσιάστηκαν, εξετάστηκαν και αναλύθηκαν η δομή και οι αρχές λειτουργίας των Ε.Σ.

Αρχικά, δόθηκε ο ορισμός του Ε.Σ. και αναφέρθηκαν τα επιμέρους τμήματα που το απαρτίζουν. Στη συνέχεια, εξετάστηκαν λεπτομερώς η δομή και η αρχιτεκτονική των Ε.Σ., δομικό στοιχείο των οποίων αποτελούν τα ψηφιακά συστήματα. Έμφαση δόθηκε στην εξέλιξη ηλεκτρονικών υπολογιστών και ιδιαίτερα σε ό,τι αφορά τη μνήμη και τους επεξεργαστές, τα οποία εμπεριέχονται σε μεγάλη ποικιλία στα Ε.Σ. Τέλος, παρουσιάστηκαν τρία χαρακτηριστικά παραδείγματα Ε.Σ. που παράγονται σε βιομηχανική κλίμακα, στα οποία συνοψίζονται και καταδεικνύονται τα χαρακτηριστικά και οι δυνατότητες των Ε.Σ. που αποτελούν αναπόσπαστο στοιχείο της καθημερινότητας του σύγχρονου ανθρώπου.

Τα κύρια συμπεράσματα που προκύπτουν από την εν λόγω πτυχιακή εργασία μπορούν να συγκεντρωθούν ως εξής:

- Τα Ε.Σ. αποτελούν ολοκληρωμένα υπολογιστικά συστήματα, τα οποία κατασκευάζονται ώστε να επιτελούν μια εξειδικευμένη λειτουργία.
- Πρόκειται για συστήματα που παρακολουθούν και ελέγχουν το περιβάλλον τους.
- Τα Ε.Σ. πολλές φορές είναι δύσκολο να καθοριστούν, διότι βρίσκονται σχεδόν κάθε υπολογιστικό σύστημα, διάφορο ενός επιτραπέζιου υπολογιστή.
- Δισεκατομμύρια μονάδες Ε.Σ. παράγονται ετησίως, έναντι των εκατομμυρίων μονάδων επιτραπέζιων υπολογιστών.

- Τα περισσότερα Ε.Σ. που απαντώνται στην καθημερινότητα είναι χαμηλού κόστους, χαμηλής ισχύος και μικρού μεγέθους.
- Τα πολύπλοκα Ε.Σ. εμπεριέχουν πολλαπλά υποσυστήματα.
- Οι κυριότεροι παράγοντες που πρέπει να λαμβάνονται υπόψη κατά τον σχεδιασμό των Ε.Σ. είναι η προβλεψιμότητα, η αξιοπιστία, η έκλυση ενέργειας, το μέγεθος, το κόστος και η απόδοση.
- Τα θέματα ασφάλειας και αξιοπιστίας είναι ζωτικής σημασίας και πρέπει να λαμβάνονται σοβαρά υπόψη κατά τον σχεδιασμό του συστήματος.
- Επειδή τα συστήματα λειτουργούν σε πραγματικό χρόνο, ο σχεδιασμός των χρονικών σταθερών της απόκρισης των μονάδων του συστήματος στα σήματα εισόδου, ο χρόνος επεξεργασίας των πληροφοριών και η δυναμική της απόκρισης αποτελούν παραμέτρους που απαιτούν βαθιά μελέτη, δοκιμή και επιβεβαίωση της απόδοσης μέχρι τον τελικό σχεδιασμό. Επιπρόσθετα, οι σχεδιαστές καλούνται να παρέχουν άνω και κάτω όρια μέσα στα οποία μπορούν να μεταβάλλονται οι παραπάνω χρόνοι του συστήματος.
- Σε ένα Λ.Σ. πραγματικού χρόνου απαιτείται μονάδα καθορισμού των προτεραιοτήτων (priorities) που αποδίδει το σύστημα κατά τη λήψη και την επεξεργασία των δεδομένων του αλλά και κατά την υλοποίηση των διεργασιών του, δηλαδή απαιτείται χρονοπρογραμματισμός των διεργασιών.
- Η αιχμή της τεχνολογίας βρίσκεται στις «ενσωματωμένες» λύσεις.
- Υπάρχει αυξημένη ανάγκη και πρόκληση βελτίωσης της τεχνολογίας των Ε.Σ., κυρίως σε επίπεδο λογισμικού.

BIBΛΙΟΓΡΑΦΙΑ

- [1] Μηνάς Δασυγένης, Δημήτριος Σούντρης, *Ενσωματωμένα Συστήματα: Ο αθέατος ψηφιακός κόσμος*, ΣΥΝΔΕΣΜΟΣ ΕΛΛΗΝΙΚΩΝ ΑΚΑΔΗΜΑΪΚΩΝ ΒΙΒΛΙΟΘΗΚΩΝ, Εθνικό Μετσόβιο Πολυτεχνείο, 2015.
- [2] Μηνάς Δασυγένης, *Εισαγωγικές έννοιες στα ενσωματωμένα συστήματα. Ορισμός. Χαρακτηριστικά. Εφαρμογές*, Πανεπιστήμιο Δυτικής Μακεδονίας, Τμήμα Μηχανικών Πληροφορικής & Τηλεπικοινωνιών.
- [3] Γρηγόρης Δουμένης, *Ενσωματωμένα Συστήματα*, ΤΕΙ ΗΠΕΙΡΟΥ-ΣΤΕΦ, ΤΜΗΜΑ ΜΗΧ. ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.
- [4] <https://physicsgg.me/2014/11/21/η-ανακάλυψη-του-τρανζίστορ/>
- [5] William Stallings, *Οργάνωση και Αρχιτεκτονική Υπολογιστών*, Εκδόσεις ΤΖΙΟΛΑ, 6^η Έκδοση, Θεσσαλονίκη 2011.
- [6] Αρετάκη Αικατερίνη, *Η UML στην ανάπτυξη ενσωματωμένων συστημάτων*, Διπλωματική εργασία, Πάτρα 2009.
- [7] <http://math.mit.edu/~rothvoss/18.304.3PM/Presentations/1-Melissa.pdf>
- [8] <http://users.uoa.gr/~akolovou/VHDL-Introduction.pdf>
- [9] <https://en.wikipedia.org/wiki/Verilog>
- [10] <http://www.embecosm.com/appnotes/ean1/html/ch02s01.html>
- [11] <https://en.wikipedia.org/wiki/SystemVerilog>
- [12] <https://www.mathworks.com/products/matlab.html>
- [13] https://en.wikipedia.org/wiki/Electronic_design_automation
- [14] <https://electronicpull.blogspot.gr/2016/05/risc-and-cisc-architectures.html>
- [15] <http://arch.icte.uowm.gr/mdasyg/book/embedded/chapter01.pdf>
- [16] Παναγιώτης Πάζιος, *Διαγνωστικά συστήματα αυτοκινήτων παραγωγής*, Διπλωματική εργασία, Πάτρα 2014

