ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
Τομέας Πυρηνικής Τεχνολογίας
Διευθυντής: Επικ. Καθηγητής Ε.Π. Χίνης

ΜΕΤΑΤΡΟΠΗ ΛΟΓΙΣΜΙΚΟΥ Λ/Σ UNIX SYSTEM V ΣΕ ΛΟΓΙΣΜΙΚΟ Λ/Σ LINUX ΓΙΑ ΤΗ ΣΥΝΔΕΣΗ ΣΕΙΡΙΑΚΩΝ ΣΥΣΚΕΥΩΝ ΜΕΣΩ ΘΥΡΑΣ RS-232

CONVERSION OF UNIX SYSTEM V SOFTWARE TO LINUX SOFTWARE FOR SERIAL DEVICES CONNECTION TO RS-232 COMPUTER PORTS

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΤΟΥ
ΑΛΕΞΟΠΟΥΛΟΥ Δ. ΓΕΩΡΓΙΟΥ

Σπουδαστή της Σχολής
Μηχανολόγων Μηχανικών ΕΜΠ

Επίβλεψη: Λέκτορας Ν.Π. Πετρόπουλος
Συνεργασία: Υποψήφιος Διδάκτορας Δ. Ι. Καράγγελος

ΑΘΗΝΑ 2008

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΕΥΧΑΡΙΣΤΙΕΣ v

ΠΕΡΙΛΗΨΗ vi

ABSTRACT viii

ΚΕΦΑΛΑΙΟ 1: **Εισαγωγή** 1-1

ΚΕΦΑΛΑΙΟ 2: **Πυρηνικά μετρητικά συστήματα με διαχωριστές – καταμετρητές**

2.1 **Εισαγωγή** 2-1

2.2 **Συγκρότηση πυρηνικών μετρητικών συστημάτων** 2-1

2.2.1 **Γενικά** 2-1

2.2.2 **Βασικές μονάδες NIM** 2-2

2.2.3 **Απλά πυρηνικά μετρητικά συστήματα με μονάδες NIM** 2-5

2.2.4 **Πυρηνικά μετρητικά συστήματα με διαχωριστές - καταμετρητές** 2-5

2.3 **Καταμετρητές CANBERRA** 2-8

2.4 **Μονάδες επικοινωνίας CANBERRA** 2-10

2.5 **Εφαρμογή στο ΕΠΤ-ΕΜΠ** 2-10

2.5.1 **Μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI** 2-11

2.5.2 **Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI** 2-12

2.6 **Συμπεράσματα - Σχόλια** 2-14

Διαγράμματα του 2^{ου} Κεφαλαίου 2-16

ΚΕΦΑΛΑΙΟ 3: **Συγκρότηση απαραίτητου υλικού, εγκατάσταση Λ/Σ LINUX και βασικού λογισμικού**

3.1 **Εισαγωγή - Ιστορικό** 3-1

3.2 **Το Λ/Σ UNIX και η εξέλιξή του στο Λ/Σ LINUX** 3-3

3.3 **Το UNIX και το LINUX στο ΕΠΤ-ΕΜΠ** 3-5

3.4 **Το Λ/Σ LINUX και ο βασικός Η/Υ της Διπλωματικής Εργασίας** 3-7

3.5 **Σειριακή θύρα** 3-12

3.6 **Επιτήρηση ASCII χαρακτήρων σειριακής επικοινωνίας** 3-17

3.7 **Συμπεράσματα - Σχόλια** 3-22

Διαγράμματα του 3^{ου} Κεφαλαίου 3-24

Πίνακες του 3^{ου} Κεφαλαίου 3-35

ΚΕΦΑΛΑΙΟ 4: **Ρύθμιση και σύνδεση σειριακής θύρας Η/Υ κάτω από το Λ/Σ LINUX**
4.1 **Εισαγωγή** 4-1
4.2 **Λ/Σ LINUX και περιφερειακές συσκευές** 4-1
4.3 **Κλήσεις συστήματος για τον έλεγχο σειριακής θύρας RS-232** 4-3
4.3.1 **Γενικά** 4-3
4.3.2 **Κλήση συστήματος open** 4-3
4.3.3 **Κλήση συστήματος read** 4-4
4.3.4 **Κλήση συστήματος write** 4-5
4.3.5 **Κλήση συστήματος close** 4-6
4.3.6 **Κλήση συστήματος lockf** 4-6
4.3.7 **Κλήση συστήματος ioctl** 4-7
4.4 **Συναρτήσεις βιβλιοθήκης POSIX** 4-7
4.4.1 **Γενικά** 4-7
4.4.2 **Έλεγχος σειριακής θύρας RS-232 μέσω της συνάρτησης termios.h** 4-8
4.4.3 **Χρήση των συναρτήσεων unistd.h, fcntl.h και sys/time.h** 4-15
4.5 **Προγραμματιστική υλοποίηση από τον Δ. Χαρβάτη (1990)** 4-15
4.5.1 **Γενικά** 4-15
4.5.2 **Κώδικας** 4-17
4.6 **Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας** 4-23
4.6.1 **Γενικά** 4-23
4.6.2 **Κώδικας** 4-24
4.7 **Ανάλυση κώδικα** 4-31
4.7.1 **Εισαγωγή** 4-31
4.7.2 **OpenCnt** 4-32
4.7.3 **SetCnt** 4-34
4.7.4 **C1earCnt** 4-36
4.7.5 **C1oseCnt** 4-37
4.7.6 **ToCnt** 4-38
4.7.7 **FromCnt** 4-38
4.7.8 **DataCnt** 4-39
4.7.9 **Time_to_I** 4-41
4.7.10 **Ellapsed** 4-41
4.8 **Χρήση κώδικα** 4-43

4.8.1 **Γενικά** 4-43
4.8.2 **Κλήση από FORTRAN** 4-43
4.8.3 **Εντολές μετάφρασης** 4-50
4.8.4 **Συγκρότηση βιβλιοθήκης** 4-51
4.9 **Συμπεράσματα - Σχόλια** 4-53
Διαγράμματα του 4^{ου} Κεφαλαίου 4-55

ΚΕΦΑΛΑΙΟ 5: **Σύνδεση σειριακής θύρας RS-232 Η/Υ με σειριακές συσκευές CANBERRA**
5.1 **Εισαγωγή** 5-1
5.2 **Συγκρότηση πειραματικής διατάξεως** 5-1
5.3 **Μονάδα επικοινωνίας 1788** 5-3
5.3.1 **Γενικά** 5-3
5.3.2 **Ρύθμιση** 5-4
5.3.3 **Πρωτόκολλο επικοινωνίας** 5-6
5.3.4 **Σύνδεση με την πειραματική διάταξη** 5-10
5.3.5 **Προγραμματιστική υλοποίηση από τον Δ. Χαρβάτη (1990)** 5-10
5.3.6 **Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας** 5-17
5.3.7 **Μετάφραση και σύνδεση** 5-34
5.4 **Μονάδα επικοινωνίας 2088** 5-34
5.4.1 **Γενικά** 5-34
5.4.2 **Ρύθμιση** 5-35
5.4.3 **Πρωτόκολλο επικοινωνίας** 5-37
5.4.4 **Σύνδεση με την πειραματική διάταξη** 5-45
5.4.5 **Προγραμματιστική υλοποίηση από τον Σιμόπουλο (1994)** 5-45
5.4.6 **Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας** 5-51
5.4.7 **Μετάφραση και σύνδεση** 5-59
5.5 **Καταμετρητής 2071A/w03 με μονάδα επικοινωνίας 207x-03** 5-59
5.5.1 **Γενικά** 5-59
5.5.2 **Ρύθμιση** 5-59
5.5.3 **Πρωτόκολλο επικοινωνίας** 5-61
5.5.4 **Σύνδεση με την πειραματική διάταξη** 5-65
5.5.5 **Προγραμματιστική υλοποίηση από τον Σιμόπουλο (1997)** 5-66
5.5.6 **Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας** 5-66

5.5.7 **Μετάφραση και σύνδεση** 5-74
5.6 **Καταμετρητής 2071Aw/02 (Daisy Chain)** 5-74
5.7 **Πρόγραμμα FORTRAN P17887.F** 5-75
5.8 **Συγκρότηση βιβλιοθήκης και συμπλήρωση makefile** 5-77
5.9 **Συμπεράσματα - Σχόλια** 5-82
Διαγράμματα του 5^{ου} Κεφαλαίου 5-83

ΕΠΙΛΟΓΟΣ Ε-1

ΒΙΒΛΙΟΓΡΑΦΙΑ Β-1

ΠΑΡΑΡΤΗΜΑ Π-1

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα Λέκτορα κ. Ν.Π. Πετρόπουλο για τη βοήθειά του καθόλη τη διάρκεια της εκπόνησης της Διπλωματικής Εργασίας και ειδικά για τη συμβολή του στη διαμόρφωση του τελικού κειμένου. Επίσης θα ήθελα να ευχαριστήσω τον Διευθυντή του Τομέα Πυρηνικής Τεχνολογίας Επίκουρο Καθηγητή κ. Ε.Π. Χίνη για τη διάθεση των απαραίτητων Η/Υ και συσκευών καθώς και για τη διάθεση αναγκαίων μικροποσών για δαπάνες προκειμένου για την πραγματοποίηση των αναγκαίων δοκιμών και το προσωπικό του Τομέα Πυρηνικής Τεχνολογίας για τη τεχνική βοήθεια και την συμπαράστασή τους, όπου χρειάσθηκε. Τέλος, οι ευχαριστίες επεκτείνονται και στον Υποψήφιο Διδάκτορα και υπάλληλο ΕΜΠ του Τομέα κ. Δ. Καράγγελο, για την συνεργασία του και τη βοήθειά του στην ανάπτυξη του λογισμικού που χρησιμοποιήθηκε στα πλαίσια της παρούσης Διπλωματικής Εργασίας.

ΠΕΡΙΛΗΨΗ

ΜΕΤΑΤΡΟΠΗ ΛΟΓΙΣΜΙΚΟΥ Λ/Σ UNIX SYSTEM V ΣΕ ΛΟΓΙΣΜΙΚΟ Λ/Σ LINUX ΓΙΑ ΤΗ ΣΥΝΔΕΣΗ ΣΕΙΡΙΑΚΩΝ ΣΥΣΚΕΥΩΝ ΜΕΣΩ ΘΥΡΑΣ RS-232

Διπλωματική Εργασία του Γ. Αλεξόπουλου

Τα τυπικά εργαστηριακά πυρηνικά μετρητικά συστήματα που χρησιμοποιούνται για την ανίχνευση, και την ποσοτική εκτίμηση ραδιενεργών ακτινοβολιών συνήθως διακρίνονται στις ακόλουθες βασικές συνιστώσες: (α) ανιχνευτής και τροφοδοτικό υψηλής τάσης, (β) προενισχυτής, (γ) ενισχυτής, (δ) διαχωριστής και τέλος (ε) ψηφιακός καταμετρητής / χρονόμετρο. Στην αρχή της ανάπτυξης αυτών των συστημάτων (~1955 – 1965) η ανάγνωση του αποτελέσματος της μέτρησης ραδιενεργού ακτινοβολίας από τον καταμετρητή γινόνταν απευθείας από τον άνθρωπο. Σύντομα, όπως ήταν φυσικό, γεννήθηκε η ανάγκη της αυτόματης καταγραφής τέτοιων αποτελεσμάτων. Ως πρώτη λύση (~1965 – 1975), χρησιμοποιήθηκαν κατάλληλοι εκτυπωτές απευθείας συνδεδεμένοι με τους καταμετρητές. Η ραγδαία ανάπτυξη των Η/Υ, έδωσε σύντομα και τη δυνατότητα οι καταγραφές αυτές να γίνονται στα μαγνητικά μέσα των πρώτων υπολογιστών (από περίπου το 1975 και μετά). Για το σκοπό αυτό χρησιμοποιήθηκαν διάφορα πρωτόκολλα επικοινωνίας με τους καταμετρητές, το χαρακτηριστικότερο από τα οποία είναι το σειριακό πρωτόκολλο επικοινωνίας που σχεδιάσθηκε ειδικά για την εξυπηρέτηση βιομηχανικών εφαρμογών με σκοπό τον έλεγχο διεργασιών και τη συλλογή δεδομένων από μετρητικές συσκευές. Με βάση τα χαρακτηριστικά του πρωτοκόλλου αυτού, η ανάγνωση και η αποθήκευση των δεδομένων από καταμετρητές, μπορεί να γίνεται on-line. Στο Εργαστήριο Πυρηνικής Τεχνολογίας ΕΜΠ (ΕΠΤ-ΕΜΠ), ήδη από τις αρχές της δεκαετίας του 1980, έγινε η προμήθεια καταμετρητών με δυνατότητες σειριακής επικοινωνίας με Η/Υ. Οι καταμετρητές αυτοί τοποθετήθηκαν σε πειραματικές διατάξεις και συνδέθηκαν με κατάλληλους Η/Υ, χρησιμοποιώντας κατάλληλο λογισμικό επικοινωνίας που αναπτύχθηκε εσωτερικά. Μάλιστα η λειτουργία των υπόψιν πυρηνικών μετρητικών συστημάτων αποδείχθηκε πολύ παραγωγική κατά τις μετρήσεις ελληνικών δειγμάτων επιφανειακού χώματος αμέσως μετά το πυρηνικό ατύχημα στο Chernobyl (Απρίλιος, 1986). Παρόλαυτά, η λειτουργία τέτοιων πυρηνικών μετρητικών συστημάτων μπορεί να απαξιώνεται ή να υπονομεύεται όχι τόσο από την τεχνολογία των επί μέρους συνιστωσών τους, αλλά κυρίως από τη σύντομη ανάγκη

αντικατάστασης των συνεργαζόμενων Η/Υ, για λόγους αναβάθμισης λογισμικού και αντικατάστασης υλικού. Αυτό το γεγονός επιβάλλει τη συχνή αναπροσαρμογή ή ακόμα και αντικατάσταση του λογισμικού επικοινωνίας των καταμετρητών με τους συνεργαζόμενους Η/Υ. Σε ό,τι αφορά στο ΕΠΤ-ΕΜΠ, η ανάγκη για τέτοιες μεγάλες αναπροσαρμογές παρουσιάσθηκε το διάστημα 1987 – 1990, κατά το οποίο αντικαταστάθηκαν τότε διαθέσιμα συστήματα Η/Υ DEC τύπου PDP-11/04 και τύπου LSI-11 από συστήματα HP-9000/320-370 με Λ/Σ UNIX. Τα συστήματα αυτά εξυπηρέτησαν –μεταξύ άλλων- και την on-line ανάγνωση πειραματικών διατάξεων μέτρησης ραδιενέργειας με καταμετρητές μέχρι και το 2007. Το 2007 η ανάγκη αυτή επανεμφανίσθηκε καθόσον τα υπολογιστικά συστήματα HP-9000/320-370 του ΕΠΤ-ΕΜΠ, ξεκίνησαν να αντικαθιστώνται σταδιακά με συστήματα Η/Υ PC συμβατά με Λ/Σ LINUX. Στην παρούσα Διπλωματική Εργασία αναπτύσσονται και δοκιμάζονται οι απαραίτητες αλλαγές στο σχετικό λογισμικό επικοινωνίας.

ABSTRACT

CONVERSION OF UNIX SYSTEM V SOFTWARE TO LINUX SOFTWARE FOR SERIAL DEVICES CONNECTION TO RS-232 COMPUTER PORTS

Diploma Dissertation of G. Alexopoulos

The typical nuclear measurement systems, which are in use in Laboratories for the detection and the quantitative evaluation of radioactive sources, consist usually out of the following basic components: (a) detector and high voltage power supply, (b) preamplifier, (c) amplifier, (d) discriminator and (e) digital counter/timer. Over the early years of the development of such measurement systems (~1955 – 1965), reading of radiation measurement results from the counters was performed by the system users themselves. However, rather soon, there was developed the need of automatically recording of such readings. Adequate printers were then(~1965 – 1975) introduced as a solution; These printers were effectively connected to appropriate counters. Rapid computer growth, in the meaning time, provided capabilities of recording such results in computer magnetic media (approx. from 1975 and on). To this end, a number of communication protocols have been informally tested for collecting counter readings, most characteristic of which, was the RS-232 serial protocol. This last protocol was specifically designed for industrial applications, in order to control processes and collect data from measuring devices. The serial protocol allows for on-line reading and recording data from counters. The Nuclear Engineering Laboratory of NTUA (NEL-NTUA) has been equipped with counters with serial communications capabilities, by the early eighties. These counters were implemented in experimental set-ups and adequately connected to appropriate computers, by using an in-hose developed communications software. The operation of the relevant nuclear measuring systems have been proven highly productive for the fast of Greek surface soil samples radioactivity, just after the Chernobyl nuclear accident (April, 1986). Nevertheless, the operation of such systems may be gradually down rated, not because of the aging technology of the nuclear components, but mainly on the ground of the rather continuous need of computer hardware and software upgrade. This fact forces for the frequent adjustment or even replacement of the counter serial communications software to the associated computer

systems. In NEL-NTUA, such major changes for this software were introduced between 1987 to 1990, during the Laboratory's transition from the DEC/PDP-11/04 and LSI-11 computer systems to the HP-9000/320-370 systems under UNIX. These Hewlett Packard systems supported inter alia the on-line recording of results from counters in nuclear measuring systems up to year 2007. In 2007 the need for changing or adjusting the communications software emerged once more, when the HP-9000/320-370 systems were scheduled for retirement. In their place PC compatible computers were selected operating under LINUX. This Dissertation deals with the introduction and testing of the necessary adjustments to the communications software.

ΚΕΦΑΛΑΙΟ 1
ΕΙΣΑΓΩΓΗ

Στόχος της παρούσας Διπλωματικής Εργασίας (ΔΕ) είναι η μετατροπή λογισμικού Λ/Σ UNIX SystemV σε λογισμικό Λ/Σ LINUX για τη σύνδεση σειριακών συσκευών μέσω θύρας RS-232. Η ΔΕ αναπτύσσεται σε έκταση πέντε Κεφαλαίων. Στο εδώ 1^{ο} Κεφάλαιο γίνεται μία σύνοψη του περιεχομένου των Κεφαλαίων 2 έως και 5.

Στο 2^{ο} Κεφάλαιο της Διπλωματικής Εργασίας αναλύονται τα πυρηνικά μετρητικά συστήματα στα οποία υπάρχουν και λειτουργούν διαχωριστές και καταμετρητές. Για το σκοπό αυτό κατ'αρχήν δίνονται ορισμοί και γίνονται βασικές περιγραφές των κυριοτέρων πυρηνικών ηλεκτρονικών τυποποίησης NIM, που ενδιαφέρουν, δηλαδή των ανιχνευτών, των τροφοδοτικών υψηλής τάσης, των προενισχυτών, των ενισχυτών, των διαχωριστών και των καταμετρητών. Σημειώνεται επίσης ότι τα πυρηνικά ηλεκτρονικά του Εργαστηρίου Πυρηνικής Τεχνολογίας ΕΜΠ (ΕΠΤ-ΕΜΠ) είναι κυρίως της κατασκευάστριας εταιρείας CANBERRA των Η.Π.Α. Παρουσιάζεται ακόμα ο τρόπος ηλεκτρικής τροφοδοσίας των μονάδων αυτών με τυποποιημένες τάσεις συνεχούς ρεύματος. Στη συνέχεια εξετάζονται με τη βοήθεια παραδειγμάτων απλές συγκροτήσεις πυρηνικών μετρητικών συστημάτων με τις μονάδες αυτές. Στο πλαίσιο της εξέτασης αυτής αναδεικνύεται και το βασικό πρόβλημα της συλλογής αποτελεσμάτων από τους καταμετρητές αυτόματα και χωρίς την ανθρώπινη παρέμβαση. Στο ΕΠΤ-ΕΜΠ, το πρόβλημα αυτό αντιμετωπίζεται με τη χρήση κατάλληλων καταμετρητών και μονάδων επικοινωνίας, ο συνδυασμός των οποίων επιτρέπει τόσο τη συλλογή δεδομένων από τους καταμετρητές, όσο και το συνολικό έλεγχο των πυρηνικών μετρητικών συστημάτων με τη χρήση σειριακής θύρας RS-232 Η/Υ. Στο ΕΠΤ-ΕΜΠ οι εφαρμογές πυρηνικών μετρητικών συστημάτων με διαχωριστές και καταμετρητές είναι κυρίως δύο: (α) μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI και (β) μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI. Καταγράφεται ότι οι εφαρμογές αυτών των συστημάτων είναι παραγωγικές και λειτουργούν απλά, αυτοματοποιημένα, εξαιρετικά οικονομικά και με αξιοπιστία. Τα συστήματα αυτά όμως απαιτούν περιοδική συντήρηση και αναβάθμιση, κυρίως διότι συστηματικά απαξιώνονται τα υπολογιστικά συστήματα και τα αντίστοιχα λειτουργικά συστήματα, που υποστηρίζουν την συλλογή δεδομένων, λόγω της συνεχούς εξέλιξης σε αυτόν τον τομέα. Το διάστημα εκπόνισης της παρούσας ΔΕ είχε ήδη διαμορφωθεί αυτή

η ανάγκη με επιτακτικό τρόπο για αναβάθμιση των συστημάτων αυτών, τόσο σε επίπεδο υλικού, όσο και σε επίπεδο λογισμικού κάτω από το λειτουργικό σύστημα LINUX.

Στο επόμενο 3ο Κεφάλαιο προτείνονται τόσο οι κατάλληλες εκδόσεις λογισμικού (στα αγγλικά software), όσο και υλικό (στα αγγλικά hardware) που είναι πρόσφορα για αυτήν την απαραίτητη αναβάθμιση. Για το σκοπό αυτό πραγματοποιείται κατ' αρχήν μία ιστορική αναδρομή πάνω στα υπολογιστικά συστήματα που διαδοχικά χρησιμοποιήθηκαν για την ανάγνωση και τον έλεγχο πυρηνικών συστημάτων με καταμετρητές στο ΕΠΤ-ΕΜΠ. Στη συνέχεια περιγράφεται η τρέχουσα (2007) κατάσταση. Στην κατάσταση αυτή το λογισμικό που χρησιμοποιείται αναπτύχθηκε σε γλώσσες προγραμματισμού C και FORTRAN από τον Χαρβάτη (1990) έτσι ώστε δύο Η/Υ τύπου HP-9000/320-370 να μπορούν να συνδεθούν με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας. Αυτό το ειδικό λογισμικό σειριακών επικοινωνιών είναι που πρέπει να αναβαθμισθεί ώστε Η/Υ PC συμβατοί με Λ/Σ LINUX να μπορούν να συνδεθούν με τους κατάλληλους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας. Για λόγους καλύτερης κατανόησης των αναγκαίων αλλαγών και μετατροπών στο λογισμικό αφιερώνονται ειδικές παράγραφοι στην εξέλιξη από το Λ/Σ UNIX στο Λ/Σ LINUX το διάστημα 1985 – 2005 και στις εφαρμογές που έχει ήδη το Λ/Σ LINUX στο ΕΠΤ-ΕΜΠ. Δίνονται επίσης λεπτομερείς πληροφορίες τόσο για την έκδοση Fedora Core 8 του Λ/Σ LINUX που χρησιμοποιείται για την προσαρμογή του ειδικού λογισμικού, καθώς και ικανή περιγραφή Η/Υ PC συμβατού που εμπλέκεται. Για την καλύτερη απόδοση του προς μετατροπή λογισμικού αναφέρεται και η εγκατάσταση στο σύστημα γλώσσας FORTRAN77. Στην συνέχεια γίνεται μία περιγραφή του σκοπού δημιουργίας "θυρών επικοινωνίας" σε Η/Υ και πειραματικές διατάξεις, αναφέρονται τα διάφορα είδη "θυρών επικοινωνίας" που απαντώνται ενώ αναλύεται ο τρόπος λειτουργίας και οι προδιαγραφές μίας σειριακής θύρας RS-232 καθώς και ο τρόπος συνεργασίας σειριακών θυρών RS-232 για την μεταφορά δεδομένων. Για τις ανάγκες της σειριακής επικοινωνίας με Η/Υ, των μονάδων επικοινωνίας και των καταμετρητών CANBERRA, πρέπει γενικά ο Η/Υ να αποστέλλει προς αυτές ASCII χαρακτήρες εντολών και να λαμβάνει από αυτές ASCII χαρακτήρες μετρήσεων. Αυτή η συναλλαγή θα είναι επιτυχής όταν από τον κώδικα του Η/Υ αποστέλλονται οι σωστοί χαρακτήρες προς τη μονάδα επικοινωνίας και λαμβάνονται από αυτή οι αναμενόμενοι χαρακτήρες που αντιπροσωπεύουν τις μετρήσεις. Διαπιστώνεται στο 3ο Κεφάλαιο ότι με την

σημερινή (2008) τεχνολογία υπάρχει άμεσος τρόπος επιβεβαίωσης της ορθής επικοινωνίας μέσω της "υποκλοπής" των αποστελλόμενων και λαμβανόμενων χαρακτήρων από το ίδιο το καλώδιο που εξυπηρετεί τη σειριακή επικοινωνία με τη χρήση κατάλληλου λογισμικού και Η/Υ "υποκλοπής". Παρουσιάζεται ως εκ τούτου και η σχετική διάταξη "υποκλοπής" που αναπτύχθηκε για την επιτήρηση της σειριακής επικοινωνίας στις εφαρμογές που ενδιαφέρουν.

Στο 4^{ο} Κεφάλαιο εξετάζεται το λογισμικό ρύθμισης, αποστολής και λήψης δεδομένων για σειριακή θύρα, όπως αναπτύχθηκε για το Λ/Σ UNIX System V από τον Χαρβάτη (1990), το οποίο συντάχθηκε σε γλώσσα προγραμματισμού C και καλεί υπορουτίνες βιβλιοθήκης του Λ/Σ UNIX System V. Το λογισμικό αυτό διακρίνεται σε τρεις ενότητες υποπρογραμμάτων: (α) τα υποπρογράμματα ρύθμισης της σειριακής θύρας, (β) τα υποπρογράμματα επικοινωνίας με τις μονάδες επικοινωνίας CANBERRA και (γ) τα γενικά υποπρογράμματα. Το σύνολο των υποπρογραμμάτων συγκροτήθηκε σε κατάλληλη βιβλιοθήκη, ώστε να διευκολυνθεί η χρήση τους και η κλήση τους από προγράμματα γραμμένα σε γλώσσα FORTRAN. Στο Κεφάλαιο αυτό παρουσιάζονται επίσης διαδοχικά ένα προς ένα όλα τα πιοπάνω υποπρογράμματα, όπως μετατράπηκαν προκειμένου να λειτουργούν κάτω από το Λ/Σ LINUX με Η/Υ τεχνολογίας PC συμβατού, με τον ίδιο ακριβώς τρόπο που λειτουργούσαν κάτω από το Λ/Σ UNIX System V με Η/Υ HP 9000. Τα υποπρογράμματα μάλιστα προσαρμόσθηκαν με τρόπο ώστε οι καλούμενες συναρτήσεις βιβλιοθήκης να είναι τυποποιημένες κατά το πρότυπο POSIX. Το σύνολο του νέου προσαρμοσμένου λογισμικού αναλύεται με λεπτομέρειες προκειμένου να αναδειχθούν τυχόν "λεπτά" σημεία. Ιδιαίτερη σημασία δίνεται επιπλέον στον τρόπο χρήσης του λογισμικού και ειδικότερα στον τρόπο που αυτό καλείται από προγράμματα εφαρμογής σε γλώσσα FORTRAN77. Σχολιάζονται επίσης οι εντολές μετάφρασης και σύνδεσης, η παραγωγή των αντίστοιχων κατάλληλων εκτελεσίμων και η συγκρότηση του αναγκαίου "makefile".

Στο 5^{ο} Κεφάλαιο ελέγχεται και χρησιμοποιείται στην πράξη το λογισμικό ρύθμισης, αποστολής και λήψης δεδομένων για σειριακή θύρα, όπως μετατράπηκε για το Λ/Σ LINUX και παρουσιάσθηκε αναλυτικά στο 4^{ο} Κεφάλαιο. Οι εφαρμογές για τις οποίες γίνεται ο σχετικός έλεγχος και η χρήση αφορούν το σύνολο των κατάλληλων καταμετρητών που υπάρχουν στο ΕΠΤ-ΕΜΠ για τους οποίους μπορεί να οργανωθεί επικοινωνία με Η/Υ μέσω σειριακής θύρας. Τα σχετικά προγράμματα εφαρμογής σε

γράφθηκαν σε γλώσσα προγραμματισμού FORTRAN77. Για την αναπαράσταση της πειραματικής πράξης χρησιμοποιήθηκε πειραματική διάταξη που μπορεί να προσομοιάσει είτε μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI, είτε μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI. Στην πειραματική διάταξη ως πηγή γεγονότων που προσομοιάζουν ραδιενεργές διασπάσεις χρησιμοποιήθηκε ένας "παλμοδότης τυχαίου θορύβου". Ο έλεγχος και η χρήση του λογισμικού ρύθμισης έγινε με πολύ επιτυχία. Τα προγράμματα εφαρμογής συνεργάσθηκαν απρόσκοπτα με τις αναγκαίες υπορουτίνες σε γλώσσα προγραμματισμού C και έτσι διαπιστώθηκε η επιτυχής και πλήρης μετατροπή λογισμικού Λ/Σ UNIX SystemV σε λογισμικό Λ/Σ LINUX για τη σύνδεση σειριακών συσκευών μέσω θύρας RS-232.

ΚΕΦΑΛΑΙΟ 2
ΠΥΡΗΝΙΚΑ ΜΕΤΡΗΤΙΚΑ ΣΥΣΤΗΜΑΤΑ ΜΕ ΔΙΑΧΩΡΙΣΤΕΣ - ΚΑΤΑΜΕΤΡΗΤΕΣ

2.1 Εισαγωγή

Ως "Πυρηνικά Ηλεκτρονικά" (στα αγγλικά Nuclear Instrumentation) ορίζονται στα πλαίσια της παρούσης Διπλωματικής Εργασίας, οι ηλεκτρονικές μονάδες που χρησιμεύουν για την προσαρμογή (ενίσχυση και διαμόρφωση) και την επεξεργασία και την καταγραφή των ηλεκτρικών σημάτων που παράγονται από ανιχνευτές ιοντιζουσών ακτινοβολιών. Με την εδώ έννοια οι ανιχνευτές αυτοί λογίζονται ως μεταλλάκτες, οι οποίοι μετατρέπουν την ενέργεια που αποδίδεται σε αυτούς όταν προσβληθούν από φωτόνιο, -α σωματίδιο, -β σωματίδιο και νετρόνιο, σε στιγμιαία (ως παλμός) μεταβολή ηλεκτρικού χαρακτηριστικού. Ονομαστικά οι ανιχνευτές αυτοί χωρίζονται στους εξής κυριότερους τύπους: (α) ανιχνευτές αερίου, (β) ανιχνευτές σπινθηρισμού και (γ) ημιαγωγούς ανιχνευτές, και διαμορφώνονται κατασκευαστικά ανάλογα και με το είδος της ακτινοβολίας που πρέπει να ανιχνευθεί (βλ. και Λεωνίδου, 1990 και CANBERRA, Edition 12, σελ. 2-7). Οι κυριότερες μονάδες Πυρηνικών Ηλεκτρονικών είναι: (α) τροφοδοτικά υψηλής τάσης, (β) οι προενισχυτές, (γ) οι ενισχυτές, (δ) οι διαχωριστές (μονοκαναλικοί αναλυτές), (ε) οι καταμετρητές, (στ) οι αναλογοψηφιακοί μετατροπείς, (ζ) οι πολυκαναλικοί αναλυτές και (η) οι μονάδες υποδοχής και ηλεκτρικής τροφοδοσίας των παραπάνω. Οι κατάλληλοι συνδυασμοί των παραπάνω Πυρηνικών Ηλεκτρονικών με ανιχνευτές συγκροτούν Πυρηνικά Μετρητικά Συστήματα

2.2 Συγκρότηση πυρηνικών μετρητικών συστημάτων

2.2.1 Γενικά

Η βιομηχανία πυρηνικών ηλεκτρονικών έχει από καιρό τυποποιήσει τη μορφή και το επιτρεπόμενο διάστημα τάσεων που πρέπει να έχουν οι επεξεργασμένοι / διαμορφωμένοι παλμοί που λαμβάνονται από τους ανιχνευτές. Έχει επίσης τυποποιήσει τις αναγκαίες τάσεις τροφοδοσίας των Μονάδων Πυρηνικών Ηλεκτρονικών (στα αγγλικά Nuclear Instrumentation Modules – NIM) καθώς και τις φυσικές διαστάσεις των αντίστοιχων ηλεκτροστατικών κουτιών (βλ. και φωτογραφίες στο Διάγραμμα 2.1). Η τυποποίηση αυτή πραγματοποιήθηκε προκειμένου οι χρήστες

να έχουν τις δυνατότητες: (α) άμεσης αντικατάστασης μονάδων με βλάβη χρησιμοποιώντας αντίστοιχες του ίδιου ή άλλου κατασκευαστή, (β) επανασυγκρότησης του μετρητικού συστήματος από τις συνιστώσες του NIM και (γ) αναβάθμισης του μετρητικού συστήματος με την προσθήκη νέων συνιστωσών NIM. Οι κυριότεροι κατασκευαστές μονάδων NIM είναι ιστορικά οι ακόλουθες:

1. ORTEC των Ηνωμένων Πολιτειών, ιδρύθηκε το 1960
2. TENNELEC των Ηνωμένων Πολιτειών, ιδρύθηκε ~ 1960
3. CANBERRA των Ηνωμένων Πολιτειών, ιδρύθηκε το 1965, ενσωμάτωσε μέρος της TENNELEC ~ 2000
4. SILENA Ιταλίας, ιδρύθηκε το 1974

Το Εργαστήριο Πυρηνικής Τεχνολογίας του ΕΜΠ (ΕΠΤ-ΕΜΠ) διαθέτει Πυρηνικά Ηλεκτρονικά, κυρίως της εταιρίας κατασκευής CANBERRA. Στη συνέχεια θα παρουσιασθούν οι βασικές μονάδες NIM, οι σημαντικότερες αρχές λειτουργίας τους και απλές συνθέσεις τους για συνήθεις μετρήσεις ιοντιζουσών ακτινοβολιών.

2.2.2 Βασικές μονάδες NIM

Το τροφοδοτικό υψηλής τάσης (στα αγγλικά high voltage unit ή HV), είναι μονάδα NIM, η οποία παρέχει συνεχή τάση, συνήθως μεγαλύτερη από 500 V, που χρησιμοποιείται για τη διατήρηση ηλεκτρικού πεδίου στο εσωτερικό των ανιχνευτών. Κατά κανόνα ιοντίζουσα ακτινοβολία π.χ. ένα φωτόνιο, που αλληλεπιδρά με την ύλη του ανιχνευτή, αλλοιώνει στιγμιαία, το υπόψιν ηλεκτρικό πεδίο με αποτέλεσμα τη δημιουργία μικρού παλμού ηλεκτρικού σήματος τάσης, ο οποίος μπορεί να μετρηθεί μετά από κατάλληλη επεξεργασία και διαμόρφωση.

Ο προενισχυτής (στα αγγλικά preamplifier) είναι μονάδα NIM, η οποία δεν έχει γενικά τυποποιημένες διαστάσεις (βλ. και φωτογραφία στο Διάγραμμα 2.2). Τοποθετείται πολύ κοντά στον ανιχνευτή ώστε να αποφεύγεται η αλλοίωση των παραγόμενων ηλεκτρικών σημάτων τάσεων σε καλώδια μεταφοράς. Το επεξεργασμένο σήμα τάσεως που λαμβάνεται από την έξοδο ενός προενισχυτή, έχει την τυπική μορφή που παρουσιάζεται στο Διάγραμμα 2.3.

Ο ενισχυτής (στα αγγλικά amplifier) είναι μονάδα NIM, με τυποποιημένες διαστάσεις (βλ. και φωτογραφία ενισχυτή στο Διάγραμμα 2.4). Τοποθετείται μετά τον προενισχυτή για να ενισχύσει περαιτέρω το ηλεκτρικό σήμα τάσης και να του δώσει

κατάλληλη διαμόρφωση. Το επεξεργασμένο ηλεκτρικό σήμα τάσης που λαμβάνεται από την έξοδο ενός ενισχυτή, μπορεί να έχει τις τυπικές μορφές που παρουσιάζονται στο Διάγραμμα 2.5 και μέγιστο ύψος τάσης μέχρι 10 V.

Ο διαχωριστής (στα αγγλικά discriminator) – ή όπως αλλιώς μπορεί να λέγεται ο μονοκαναλικός αναλυτής (στα αγγλικά single channel analyzer), έχει κυρίως δύο σκοπούς: (α) να διαχωρίσει τα ηλεκτρικά σήματα τάσης που προέρχονται από τον ενισχυτή σε θόρυβο και σε σήματα τάσης που οφείλονται σε πραγματική αλληλεπίδραση ιοντίζουσας ακτινοβολίας με τον ανιχνευτή, και (β) από τα σήματα τάσης που οφείλονται σε πραγματική αλληλεπίδραση ιοντίζουσας ακτινοβολίας με τον ανιχνευτή να κάνει αποδεκτά μόνο εκείνα που έχουν ύψος παλμού μέχρι μία συγκεκριμένη τιμή. Τα σήματα θορύβου έχουν πολύ χαμηλό ύψος παλμού. Ο διαχωριστής διαθέτει ρύθμιση (σε V), η οποία ονομάζεται "κατώφλι" (στα αγγλικά noise level ή lower level - LL). Σήματα τάσης με ύψος παλμού κάτω από το κατώφλι δεν είναι αποδεκτά. Επίσης ο διαχωριστής διαθέτει ρύθμιση (σε V), η οποία ονομάζεται "παράθυρο" (στα αγγλικά window - ΔV). Σήματα τάσης με ύψος παλμού πάνω από το άθροισμα κατώφλι + παράθυρο δεν είναι αποδεκτά. Για κάθε αποδεκτό σήμα τάσης ο διαχωριστής παράγει έναν λογικό παλμό τάσης (στα αγγλικά logic pulse), του οποίου η τυπική μορφή διακρίνεται στο Διάγραμμα 2.6 και το τυπικό του ύψος είναι περί τα 6 V.

Ο καταμετρητής (στα αγγλικά counter) απαριθμεί σε ψηφιακή οθόνη LED, το πλήθος των λογικών παλμών που του αποστέλλει ο διαχωριστής. Οι καταμετρητές διαθέτουν συνήθως και χρονόμετρα ώστε να μπορεί να υπολογίζεται ο αριθμός των αποδεκτών παλμών στη μονάδα του χρόνου. Γενικά, οι καταμετρητές μπορεί να είναι μίας οθόνης (counter) ή δύο οθονών (dual counter/timer). Σε αυτήν την περίπτωση στη δεύτερη οθόνη μπορεί να παρουσιάζεται ο χρόνος ή οι λογικοί παλμοί από άλλο διαχωριστή.

Στην περίπτωση που ένας ανιχνευτής έχει τη δυνατότητα να παράγει ηλεκτρικό σήμα τάσης, του οποίου η τιμή να είναι ανάλογη της ενέργειας της ιοντίζουσας ακτινοβολίας (π.χ,. ενός φωτονίου) που αλληλεπίδρασε με αυτόν, τότε χρησιμοποιώντας πολλούς διαχωριστές, διαφορετικά ρυθμισμένους ως προς το LL και το ΔV, είναι δυνατόν να απαριθμώνται σε αντίστοιχα πολλούς καταμετρητές λογικοί παλμοί που οφείλονται π.χ. σε φωτόνια διαφορετικής ενέργειας. Ως αποτέλεσμα είναι δυνατόν να εκτιμηθεί η ποσότητα π.χ. των φωτονίων διαφόρων

ενεργειών που μπορεί να εκπέμπει ένα δείγμα και έτσι να εκτιμηθεί ποια ισότοπα που εκπέμπουν φωτόνια περιέχονται σε αυτό και σε ποιες ποσότητες, δεδομένου ότι είναι γνωστό ότι κάθε ισότοπο που εκπέμπει φωτόνια, τα εκπέμπει σε συγκεκριμένες ενέργειες. Φυσικά τέτοιες πολύπλοκες συνθέσεις μονάδων NIM αποδεικνύονται ασύμφορες.

Με την πρόοδο των ηλεκτρονικών δημιουργήθηκαν οι αναλογοψηφιακοί μετατροπείς (στα αγγλικά analog-to-digital-converter ή ADC). Οι μετατροπείς αυτοί χρησιμοποιούνται ως μονάδες NIM σε συνεργασία με πολυκαναλικούς αναλυτές (στα αγγλικά multi-channel-analyzer ή MCA) ως εξής:

> Ο αναλογοψηφιακός μετατροπέας αναλαμβάνει να μετατρέψει κάθε ύψος παλμού τάσης που λαμβάνει από τον ενισχυτή σε διακριτή αριθμητική τιμή. Όσο μεγαλύτερο το ύψος του παλμού τάσης τόσο μεγαλύτερη είναι και η διακριτή αριθμητική τιμή που αποδίδεται. Για ίδια ύψη παλμών τάσεων αποδίδεται πάντα ίδια διακριτή αριθμητική τιμή. Τις διακριτές αυτές τιμές τις παραλαμβάνει στη συνέχεια ο πολυκαναλικός αναλυτής. Για κάθε μία από αυτές τις τιμές ο πολυκαναλικός αναλυτής διαθέτει ένα ηλεκτρονικό απαριθμητή, π.χ. αν η διακριτή τιμή είναι "10", ο απαριθμητής του "10" αυξάνει κατά ένα, το ίδιο και για κάθε άλλη δυνατή διακριτή τιμή. Ως αποτέλεσμα μετά το χρονικό τέλος μιας μέτρησης ενός δείγματος είναι δυνατόν να "διαβάζονται" από Η/Υ όλοι οι απαριθμητές και να εκτιμάται ο αριθμός των π.χ. φωτονίων διαφορετικών ενεργειών που μετρήθηκαν από ένα δείγμα.

Οι τυποποιημένες (κατά DOE/ER-0457/1964) μονάδες υποδοχής και τροφοδοσίας των παραπάνω μονάδων NIM, έχει επικρατήσει να ονομάζονται "δοχεία" NIM (στα αγγλικά NIM Bin ή NIM Crate). Φωτογραφίες τέτοιων μονάδων διακρίνονται στο Διάγραμμα 2.7. Οι μονάδες NIM Bin, διαθέτουν ενσωματωμένο τροφοδοτικό το οποίο έχει τη δυνατότητα να τροφοδοτεί τις μονάδες NIM που τοποθετούνται στις υποδοχές της με +24, -24, +12 και –12 V. Κατά τη νεότερη τυποποίηση DOE/ER-0457T/1990, το τροφοδοτικό των NIM Bin απέκτησε τη δυνατότητα να τροφοδοτεί τις μονάδες με επιπλέον +6 και –6 V. Στα σχήματα του Διαγράμματος 2.8 διακρίνεται η μορφή των θηλυκών (από την πλευρά της NIM Bin) και των αρσενικών (από την πλευρά των μονάδων NIM) connectors, που χρησιμεύουν για την τροφοδοσία των

μονάδων NIM όταν αυτές τοποθετηθούν μέσα στην NIM Bin. Στο σχήμα του Διαγράμματος 2.9 διακρίνεται και η τυποποιημένη χρησιμότητα κάθε υποδοχής.

2.2.3 Απλά πυρηνικά μετρητικά συστήματα με μονάδες NIM

Ένα παράδειγμα απλής διάταξης με μονάδες NIM είναι αυτό που διακρίνεται στο σχήμα του Διαγράμματος 2.10. Σε αυτό το πυρηνικό μετρητικό σύστημα η βαθμίδα ενισχυτής + διαχωριστής, χρησιμοποιείται για την παραγωγή λογικών παλμών από ηλεκτρικά σήματα τάσεων που οφείλονται σε αλληλεπιδράσεις ακτινοβολίας με τον ανιχνευτή και βρίσκονται στα πλαίσια των τιμών του αποδεκτού "παραθύρου τάσεων" (LL, LL+ΔV). Οι λογικοί παλμοί χρησιμοποιούνται ως είσοδος στον καταμετρητή / χρονόμετρο, ο οποίος μπορεί να ρυθμίζεται ώστε να σταματάει την καταγραφή όταν είτε ο χρόνος, είτε ο αριθμός των καταμετρηθέντων παλμών φθάσει σε μία προρυθμισμένη τιμή. Ένας καταμετρητής μπορεί να διαθέτει κατάλληλη θύρα επικοινωνίας (συνήθως σειριακή) ή μπορεί να συνδεθεί με μονάδα επικοινωνίας με σειριακές θύρες, ώστε είτε να τυπώνει τα αποτελέσματα απευθείας σε εκτυπωτή, είτε να τα τροφοδοτεί σε Η/Υ.

Στις σημερινές (2008) απλές διατάξεις –πλην εξαιρέσεων- ο διαχωριστής και ο καταμετρητής χρησιμοποιούνται επικουρικά. Στη θέση τους προτιμάται να χρησιμοποιούνται αναλογοψηφιακός μετατροπέας και πολυκαναλικός αναλυτής σε σύνδεση με Η/Υ, όπως διακρίνεται στο σχήμα του Διαγράμματος 2.11. Οι λόγοι για αυτό είναι οι ακόλουθοι: (α) το κόστος των μονάδων NIM διαχωριστής + καταμετρητής, είναι πλέον συγκρίσιμο με το κόστος των μονάδων αναλογοψηφιακός μετατροπέας + πολυκαναλικός αναλυτής, σε αντίθεση με το παρελθόν (1970 – 2000), και (β) η χρήση αναλογοψηφιακού μετατροπέα + πολυκαναλικού αναλυτή, επιτρέπει την ανίχνευση ιοντίζουσας ακτινοβολίας και από ενέργειες, οι οποίες δεν αναμένονται, πράγμα που δεν είναι δυνατό να γίνεται από συστήματα πολλών διαχωριστών – καταμετρητών με προρυθμισμένα ενεργειακά παράθυρα.

2.2.4 Πυρηνικά μετρητικά συστήματα με διαχωριστές - καταμετρητές

Για την ανάπτυξη του τρόπου λειτουργίας και της χρησιμότητας διατάξεων με διαχωριστές και καταμετρητές υποτίθεται σε αυτό το σημείο της Διπλωματικής Εργασίας ότι ο ανιχνευτής παράγει παλμούς τάσης το ύψος των οποίων εξαρτάται από την ενέργεια των φωτονίων που αλληλεπιδρούν με αυτόν.

Χρησιμοποιείται η διάταξη του σχήματος στο Διάγραμμα 2.10 ως εξής:

1. Ρυθμίζεται ο διαχωριστής ώστε $LL=LL_1$ να έχει κατάλληλη τιμή που να αποκλείει τον θόρυβο.
2. Ρυθμίζεται το παράθυρο ΔV σε ικανοποιητικά μικρή τιμή
3. Λαμβάνεται μέτρηση από ένα δείγμα που εκπέμπει φωτόνια για συγκεκριμένο σταθερό χρόνο
4. Καταγράφονται οι λογικοί παλμοί που καταμετρήθηκαν από τον καταμετρητή, είτε χειρονακτικά είτε σε κατάλληλο εκτυπωτή.
5. Ρυθμίζεται ο διαχωριστής ώστε $LL_{i+1}=LL_i+\Delta V$, i=1,n και επαναλαμβάνονται τα βήματα "3","4" και "5" μέχρι το i να γίνει τέτοιο που $LL_i+\Delta V=10V$ (η οποία είναι η μέγιστη τιμή τάσης που κατά την τυποποίηση των πυρηνικών ηλεκτρονικών μπορεί να δίνει ένας ενισχυτής).
6. Οι καταγραφές των βημάτων "4" παριστάνονται σε συνάρτηση με την τιμή $LL_i+\Delta V$, για κάθε i, και αντιπροσωπεύουν στην ουσία την ενεργειακή κατανομή των φωτονίων που αλληλεπίδρασαν με τον ανιχνευτή.

Πρόκειται για εξαιρετικά κοπιώδη και χρονοβόρα διαδικασία, η οποία χρησιμοποιούνταν κυρίως το διάστημα 1960 – 1970 με τις τότε διαθέσιμες μονάδες NIM, για την ποσοτική και ποιοτική αξιολόγηση δειγμάτων που περιείχαν ισότοπα που εξέπεμπαν φωτόνια.

Χρησιμοποιείται η διάταξη του σχήματος στο Διάγραμμα 2.12 με πολλά ζεύγη διαχωριστών- καταμετρητών ως εξής:

1. Ρυθμίζεται ο διαχωριστής "1" ώστε $LL=LL_1$ να έχει κατάλληλη τιμή που να αποκλείει τον θόρυβο.
2. Ρυθμίζεται το παράθυρο ΔV όλων των διαχωριστών σε ικανοποιητικά μικρή τιμή.
3. Ρυθμίζεται ο διαχωριστής "i" ώστε $LL_i=LL_{i-1}+\Delta V$, i=1,n.
4. Λαμβάνεται μέτρηση από ένα δείγμα που εκπέμπει φωτόνια για συγκεκριμένο σταθερό χρόνο

5. Καταγράφονται οι λογικοί παλμοί που καταμετρήθηκαν από τους i καταμετρητές, είτε χειρονακτικά είτε σε κατάλληλο εκτυπωτή, είτε με κατάλληλα συνδεδεμένο Η/Υ.

Πρόκειται για λιγότερο κοπιώδη και χρονοβόρα διαδικασία, η οποία χρησιμοποιούνταν κυρίως το διάστημα 1965 – 1980 με τις τότε διαθέσιμες μονάδες NIM, για την ποσοτική και ποιοτική αξιολόγηση δειγμάτων που περιείχαν ισότοπα που εξέπεμπαν φωτόνια. Η διαδικασία αυτή όμως αντιμετωπίζει το εξής κυριότερο πρόβλημα: οι καταμετρητές πρέπει να ξεκινούν ταυτόχρονα και να σταματούν ταυτόχρονα. Αυτό εξασφαλίζεται όταν όλοι οι καταμετρητές είναι συνδεδεμένοι με κατάλληλη καλωδίωση. Ένας από τους καταμετρητές ορίζεται ως "κυβερνήτης" (στα αγγλικά master) και το χρονόμετρό του μπορεί να δίνει το σήμα έναρξης και λήξης της μέτρησης στους υπόλοιπους που ορίζονται ως "εξαρτώμενοι" (στα αγγλικά slave). Η διάταξη των καταμετρητών με αυτόν τον τρόπο ονομάζεται Daisy Chain. Μέχρι 50 κατ'αρχήν καταμετρητές (με μία ή δύο οθόνες) CANBERRA, μπορούν να συνδέονται σε διάταξη Daisy Chain, σχηματική αναπαράσταση της οποίας δίνεται στο Διάγραμμα 2.13. Η καλωδίωση Daisy Chain είναι κατασκευασμένη με τέτοιο τρόπο ώστε από αυτήν να μπορούν όταν χρειάζεται να διαβιβάζονται και τα δεδομένα που καταγράφονται στους καταμετρητές προς έναν εκτυπωτή ή ένα Η/Υ. Ένα δεύτερο πρόβλημα είναι ότι το πλήθος των μετρήσεων είναι σε αυτή την περίπτωση μεγάλο και πρέπει να χρησιμοποιείται για την καταγραφή τους οπωσδήποτε κατάλληλος εκτυπωτής ή και σύνδεση με Η/Υ. Η χειρωνακτική καταγραφή δεν συμφέρει, ιδιαίτερα μάλιστα όταν ζητείται να παρακολουθηθεί η εξέλιξη της διάσπασης ισοτόπων με μικρό χρόνο ημιζωής. Σε αυτήν την περίπτωση οι καταμετρητές πρέπει αφού αποστείλουν τα δεδομένα στον εκτυπωτή ή τον Η/Υ να ξαναεκκινήσουν το συντομότερο και κατά προτίμηση αυτόματα. Τόσο οι εκτυπωτές, όσο και οι Η/Υ χρησιμοποιούν συνήθως για τέτοιες επικοινωνίες σειριακές θύρες. Επειδή η τεχνολογία Daisy Chain δεν είναι συμβατή με σειριακές επικοινωνίες απαιτείται για το σκοπό της μεταβίβασης δεδομένων από τους καταμετρητές σε εκτυπωτή ή Η/Υ, κατάλληλη μονάδα επικοινωνίας. Η μονάδα αυτή αναλαμβάνει τον έλεγχο του master καταμετρητή στην αλυσίδα Daisy Chain, έτσι ώστε όχι μόνο παθητικά να δέχεται και να αποστέλλει δεδομένα αλλά και να μπορεί επίσης να μεταβιβάζει εντολές ελέγχου στους καταμετρητές (π.χ. μηδενισμός ενδείξεων, έναρξη μέτρησης, τερματισμός

μέτρησης, ανάγνωση μέτρησης). Στο Διάγραμμα 2.14 διακρίνονται φωτογραφίες διάταξης Daisy Chain, με δύο καταμετρητές και μονάδα επικοινωνίας.

2.3 Καταμετρητές CANBERRA

Το ΕΠΤ-ΕΜΠ διαθέτει μία συλλογή από καταμετρητές / χρονόμετρα εταιρείας κατασκευής CANBERRA, ως εξής:

Μοντέλο 1772 (διατίθεται 1 μονάδα)

Κατασκευαζόταν μέχρι το 1982 (βλ. CANBERRA, 81-82, σελ. 183-184). Κύρια χαρακτηριστικά: είσοδος για θετικούς και αρνητικούς λογικούς παλμούς, ξεχωρίζει μεταξύ τους παλμούς που απέχουν χρονικά πάνω από 40 ns, δεν μπορεί να συνδεθεί με άλλον καταμετρητή, με εκτυπωτή ή Η/Υ. Δέχεται είσοδο από έναν και μόνο διαχωριστή. Οι μετρήσεις διαβάζονται από ψηφιακή οθόνη LED, που ονομάζεται "παράθυρο" (στα αγγλικά scaler), έξι χαρακτήρων (βλ. λοιπά χαρακτηριστικά στο Παράρτημα)

Μοντέλο 1776 (διατίθενται 7 μονάδες)

Κατασκευαζόταν μέχρι το 1982 (βλ. CANBERRA, 81-82, σελ. 191-192). Κύρια χαρακτηριστικά: είσοδος για θετικούς και αρνητικούς λογικούς παλμούς, ξεχωρίζει μεταξύ τους παλμούς που απέχουν χρονικά πάνω από 40 ns, μπορεί να συνδεθεί με άλλον καταμετρητή σε διάταξη Daisy Chain, με εκτυπωτή ή Η/Υ μέσω μονάδας επικοινωνίας και τεχνολογίας Daisy Chain. Μπορεί να δέχεται είσοδο από έναν ή δύο διαχωριστές (είναι δηλαδή διπλός καταμετρητής – dual counter). Οι μετρήσεις διαβάζονται από δύο ψηφιακές οθόνες LED, των έξι χαρακτήρων. Οι αντίστοιχοι ψηφιακοί καταχωρητές που οδηγούν τις οθόνες έχουν μήκος επίσης έξι χαρακτήρες (βλ. λοιπά χαρακτηριστικά στο Παράρτημα)

Μοντέλο 2071Α (διατίθενται 2 μονάδες)

Κατασκευάζονταν από το 1983 μέχρι περίπου το 2000 (βλ. CANBERRA, Edition 12, σελ. 209-210) με τα ακόλουθα κύρια χαρακτηριστικά: είσοδος για θετικούς και αρνητικούς λογικούς παλμούς, ξεχωρίζει μεταξύ τους αρνητικούς παλμούς που απέχουν χρονικά πάνω από 4 ns, ξεχωρίζει μεταξύ τους θετικούς παλμούς που απέχουν χρονικά πάνω από 40 ns. Μπορεί να δέχεται είσοδο από έναν ή δύο διαχωριστές (είναι δηλαδή διπλός καταμετρητής – dual counter). Οι μετρήσεις διαβάζονται από μία ψηφιακή οθόνη LCD, των οκτώ χαρακτήρων, η οποία

παρουσιάζει τα αποτελέσματα του πρώτου ή του δεύτερου καταμετρητή με τη χρήση ενός απλού διακόπτη (βλ. λοιπά χαρακτηριστικά στο Παράρτημα).

Μοντέλο 2071Aw/02 (διατίθεται 1 μονάδα)

Κατασκευάζονταν από το 1983 μέχρι περίπου το 1991 (βλ. CANBERRA, Edition 8, σελ. 174-175). Όπως το 2071A αλλά με δυνατότητα Daisy Chain, δυνατότητα σύνδεσης με εκτυπωτή ή Η/Υ μέσω μονάδας επικοινωνίας και τεχνολογίας Daisy Chain. Οι μετρήσεις διαβάζονται από μία ψηφιακή οθόνη LCD, των έξι χαρακτήρων, η οποία παρουσιάζει τα αποτελέσματα του πρώτου ή του δεύτερου καταμετρητή με τη χρήση ενός απλού διακόπτη. Οι αντίστοιχοι ψηφιακοί καταχωρητές που οδηγούν τις οθόνες έχουν μήκος έξι χαρακτήρες (βλ. λοιπά χαρακτηριστικά στο Παράρτημα).

Μοντέλο 2071A/w03 με ενσωματωμένη μονάδα επικοινωνίας 207x-03 (διατίθεται 1 μονάδα)

Κατασκευάζονταν από περίπου το 1986 μέχρι περίπου το 1997(βλ. CANBERRA, Edition 10, σελ. 197-199). Όπως το 2071A αλλά με δυνατότητα σύνδεσης με άλλους καταμετρητές με εκτυπωτή ή Η/Υ μέσω ενσωματωμένης σειριακής μονάδας επικοινωνίας. Οι μετρήσεις διαβάζονται από μία ψηφιακή οθόνη LCD, των έξι χαρακτήρων, η οποία παρουσιάζει τα αποτελέσματα του πρώτου ή του δεύτερου καταμετρητή με τη χρήση ενός απλού διακόπτη (βλ. λοιπά χαρακτηριστικά στο Παράρτημα). Οι αντίστοιχοι ψηφιακοί καταχωρητές που οδηγούν τις οθόνες έχουν μήκος οκτώ χαρακτήρες (βλ. λοιπά χαρακτηριστικά στο Παράρτημα).

Μοντέλο 512 (διατίθενται 2 μονάδες)

Κατασκευάζεται από το 2000 μέχρι σήμερα (βλ. http://www.canberra.com/pdf/Products/NIM_pdf/Model_512_super_spec.pdf) με τα ακόλουθα κύρια χαρακτηριστικά: είσοδος για θετικούς και αρνητικούς λογικούς παλμούς, ξεχωρίζει μεταξύ τους αρνητικούς παλμούς που απέχουν χρονικά πάνω από 4 ns, ξεχωρίζει μεταξύ τους θετικούς παλμούς που απέχουν χρονικά πάνω από 40 ns. Διαθέτει δυνατότητες επικοινωνίας και ελέγχου από Η/Υ, μέσω σειριακής θύρας RS-232 και πρωτοκόλλου GPIB IEEE-488. (βλ. λοιπά χαρακτηριστικά στο Παράρτημα). Σημειώνεται ότι αυτοί οι καταμετρητές δεν χρησιμοποιούνται αυτή τη στιγμή σε καμία διάταξη στο ΕΠΤ-ΕΜΠ και είναι διαθέσιμοι για χρήση σε περίπτωση βλάβης ενός από τα άλλα διαθέσιμα μοντέλα.

2.4 Μονάδες επικοινωνίας CANBERRA

Το ΕΠΤ-ΕΜΠ διαθέτει έναν περιορισμένο αριθμό από μονάδες επικοινωνίας, οι οποίες είναι κατάλληλες για τη μεταβίβαση δεδομένων από τους καταμετρητές σε εκτυπωτή ή Η/Υ μέσω σειριακής πόρτας. Οι μονάδες αυτές είναι κατασκευής CANBERRA, ως εξής:

Μοντέλο 1788 (διατίθεται 1 μονάδα)

Κατασκευαζόταν μέχρι το 1985 (βλ. CANBERRA, Edition 6, σελ. 155-156) με τα ακόλουθα κύρια χαρακτηριστικά: Πλήρης συμβατότητα με το σειριακό πρωτόκολλο επικοινωνίας Electronic Industries Association EIA RS-232C. Μέσω εσωτερικών jumpers: ρυθμιζόμενη ταχύτητα επικοινωνίας 300 / 1200 / 2400/ 4800/ 9600 Baud, δυνατότητα ρύθμισης parity, δυνατότητα ρύθμισης πλήθους stop bit, λειτουργία "computer" (ασύγχρονη αποστολή δεδομένων και λήψη εντολών), λειτουργία "modem" (σύγχρονη αποστολή δεδομένων και λήψη εντολών). Όλες οι επικοινωνίες με έναν Η/Υ γίνονται με τη χρήση ASCII χαρακτήρων (βλ. λοιπά χαρακτηριστικά στο Instruction Manual, 1978 - Παράρτημα).

Μοντέλο 2088 (διατίθενται 2 μονάδες)

Κατασκευάζονταν από το 1983 μέχρι περίπου το 1991 (βλ. CANBERRA, Edition 8, σελ. 180-181) με τα ακόλουθα κύρια χαρακτηριστικά: Πλήρης συμβατότητα με το σειριακό πρωτόκολλο επικοινωνίας Electronic Industries Association EIA RS-232C. Μέσω εσωτερικών jumpers: ρυθμιζόμενη ταχύτητα επικοινωνίας 110 / 150 / 300 / 600 / 1200 / 2400 / 4800/ 9600 / 19200 / 38400 Baud, δυνατότητα ρύθμισης parity, δυνατότητα ρύθμισης πλήθους stop bit, λειτουργία "computer" (ασύγχρονη αποστολή δεδομένων και λήψη εντολών), λειτουργία "modem" (σύγχρονη αποστολή δεδομένων και λήψη εντολών). Όλες οι επικοινωνίες με έναν Η/Υ γίνονται με τη χρήση ASCII χαρακτήρων(βλ. λοιπά χαρακτηριστικά στο Operator's Manual, 1987 - Παράρτημα)

2.5 Εφαρμογή στο ΕΠΤ-ΕΜΠ

Στο ΕΠΤ-ΕΜΠ οι εφαρμογές πυρηνικών μετρητικών συστημάτων με διαχωριστές και καταμετρητές είναι κυρίως δύο: (α) μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI και (β) μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI.

2.5.1 Μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI

Η ανάγκη για οικονομικές, γρήγορες και αξιόπιστες μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα (σε Bqkg^{-1}) προέκυψε στο ΕΠΤ-ΕΜΠ, το 1986 μετά το ατύχημα του Chernobyl και τη συνακόλουθη εναπόθεση ισοτόπων του ραδιενεργού νέφους στα επιφανειακά εδάφη της Ελλάδας. Όπως αναφέρεται στον Simopoulos (1989), προκειμένου να διερευνηθεί η έκταση του φαινομένου, συλλέχθηκαν την περίοδο Μάιος – Νοέμβριος 1986 1242 δείγματα επιφανειακού εδάφους σχεδόν από το σύνολο της ελληνικής ηπειρωτικής χώρας. Η ανάλυσή τους σχετικά με το περιεχόμενο ^{137}Cs έγινε ως εξής:

Οι μετρήσεις των δειγμάτων σε συγκεκριμένη γεωμετρία, έγιναν με ανιχνευτή NaI, διαστάσεων 3" x 3", μέσα σε θωράκιση μολύβδου πάχους 50 mm. Τα σήματα τάσεως από τον ανιχνευτή μετά την ενίσχυση τροφοδοτούσαν έναν διαχωριστή, στον οποίο το διάστημα (LL, LL+ΔV) είχε ρυθμισθεί ώστε να καταγράφονται μόνο παλμοί τάσης που οφείλονταν στην ενέργεια των 661.62 keV των φωτονίων του ^{137m}Ba (^{137}Cs). Ο διαχωριστής παρήγαγε αντίστοιχους λογικούς παλμούς, οι οποίοι τροφοδοτούσαν καταμετρητή Model 1776. Ο καταμετρητής ήταν συνδεδεμένος μέσω καλωδίωσης Daisy Chain με μονάδα επικοινωνίας Model 1788. Η μονάδα επικοινωνίας συνδεόταν σειριακά με δύο Η/Υ DEC τύπου PDP 11/04 και τύπου LSI-11, κάτω από το λειτουργικό σύστημα RT-11. Ο Η/Υ είχε την ευθύνη, τόσο της συλλογής των δεδομένων από τον καταμετρητή, όσο και του ελέγχου του συνόλου του συστήματος (π.χ. μηδενισμός ενδείξεων, έναρξη μέτρησης, τερματισμός μέτρησης, ανάγνωση μέτρησης) σύμφωνα με την ακόλουθη διαδικασία:

- κάθε δείγμα μετρώνταν για 20 s, τουλάχιστον i=3 και όχι περισσότερες από i=10 φορές.

- αν το τυπικό σφάλμα μιας σειράς i μετρήσεων ενός δείγματος υπολογιζόταν σε λιγότερο από 5%, τότε ακόμα και αν i=3 η μέτρηση του δείγματος θεωρούνταν ικανοποιητικής ακρίβειας και δεν συνεχίζονταν.

Το βασικό λογισμικό υλοποίησης της διαδικασίας αυτής περιγράφεται στον Σιμόπουλος (1986). Όλα τα αποτελέσματα καταγράφονταν αυτόματα σε κατάλληλη βάση δεδομένων που περιείχε επίσης λεπτομερειακές πληροφορίες σχετικά με τη γεωγραφική θέση και τη χρονική στιγμή συλλογής του δείγματος καθώς και σχετικά με τη μάζα του. Με τη χρήση αυτής της διαδικασίας, ο αναγκαίος χρόνος για τη

μέτρηση ενός δείγματος, και μάλιστα χωρίς τη χρήση αναλογοψηφιακού μετατροπέα και πολυκαναλικού αναλυτή, περιορίσθηκε πάρα πολύ, μεταξύ 60 και 200 s. Αυτό επέτρεψε την πολύ σύντομη και αξιόπιστη συμπλήρωση των μετρήσεων συγκέντρωσης ^{137}Cs στα 1242 συλλεχθέντα δείγματα και την εξαγωγή ασφαλών συμπερασμάτων σχετικά με τις πλέον επιβαρημένες περιοχές της χώρας από απόθεση ^{137}Cs εξαιτίας του ατυχήματος στο Chernobyl. Το σύστημα που περιγράφθηκε, υπάρχει συγκροτημένο και ενεργό στο ΕΠΤ-ΕΜΠ, διασφαλίζοντας και την ετοιμότητα του Εργαστηρίου σε σχέση με πιθανό ατύχημα έκλυσης ραδιοϊσοτόπων στον Ελληνικό χώρο (βλ. και φωτογραφία του Διαγράμματος 2.15). Δεδομένου ότι τα συστήματα Η/Υ DEC τύπου PDP 11/04 και LSI-11, δεν λειτουργούν πια λόγω παλαιότητας, το σύστημα εξυπηρετείται από Η/Υ τύπου PC συμβατού σειράς AT. Τη σχετική προσαρμογή του αναγκαίου λογισμικού σε Λ/Σ Microsoft DOS και γλώσσα assembly για επεξεργαστή Intel 80x86, ανέπτυξε στα πλαίσια Διπλωματικής Εργασίας ο Βοσυνιώτης (1990).

2.5.2 Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI

Η ανάγκη για οικονομικές, γρήγορες και αξιόπιστες εκτιμήσεις συγκέντρωσης ^{137}Cs (σε Bqm^{-3}) στην ατμόσφαιρα της Πολυτεχνειούπολης Ζωγράφου προέκυψε στο ΕΠΤ-ΕΜΠ, με παρόμοιο όπως στην προηγούμενη παράγραφο τρόπο, το 1986 μετά το ατύχημα του Chernobyl. Την περίοδο εκείνη δεν υπήρχαν στο ΕΠΤ-ΕΜΠ, μέσα δειγματοληψίας ρύπων στον αέρα μέσω φίλτρων, δεν υπήρχαν δηλαδή αντλίες δειγματοληψίας αέρα βαθμονομημένης παροχής και με κατάλληλες υποδοχές για χάρτινα φίλτρα ή για φίλτρα από ίνες γυαλιού (glass fiber) μεγάλης επιφάνειας. Ως εκ τούτου διαμορφώθηκε η εξής μέθοδος εκτίμησης της ανόδου της συγκέντρωσης ^{137}Cs στην ατμόσφαιρα εξαιτίας εκλύσεων από πυρηνικό ατύχημα:

Ανιχνευτής NaI, διαστάσεων 2" x 2", τοποθετείται γυμνός στον αέρα του περιβάλλοντος, μέσα σε κλωβό προστασίας (μεταλλικό ντουλαπάκι) από τις καιρικές συνθήκες δηλαδή την βροχή και τον έντονο αέρα (βλ. και φωτογραφία του Διαγράμματος 2.16). Τα σήματα τάσεως από τον ανιχνευτή μετά την ενίσχυση τροφοδοτούν έναν διαχωριστή, στον οποίο το διάστημα (LL, LL+ΔV) είχε ρυθμισθεί ώστε να καταγράφονται μόνο παλμοί τάσης που οφείλονταν στην ενέργεια των 661.62 keV των φωτονίων του ^{137m}Ba (^{137}Cs) που ενδεχόμενα βρίσκεται αιωρούμενο στην ατμόσφαιρα. Ο διαχωριστής παράγει αντίστοιχους λογικούς παλμούς, οι οποίοι

τροφοδοτούν καταμετρητή Model 1776. Ο καταμετρητής είναι συνδεδεμένος μέσω καλωδίωσης Daisy Chain με μονάδα επικοινωνίας Model 2088. Η μονάδα επικοινωνίας συνδέεται σειριακά με ένα Η/Υ HP-9000/320 με λειτουργικό σύστημα UNIX. Ο Η/Υ έχει την ευθύνη, τόσο της συλλογής των δεδομένων από τον καταμετρητή, όσο και του ελέγχου του συνόλου του συστήματος (π.χ. μηδενισμός ενδείξεων, έναρξη μέτρησης, τερματισμός μέτρησης, ανάγνωση μέτρησης) σύμφωνα με την ακόλουθη διαδικασία:

- μία μέτρηση αποτελείται από σειρά επιμέρους μετρήσεων, κάθε μία από τις οποίες διαρκεί 100 s.
- το πλήθος των επιμέρους μετρήσεων καθορίζεται από το άνω αποδεκτό όριο του τυπικού σφάλματος
- μία μέτρηση επαναλαμβάνεται υπό συνθήκες συνήθους καθημερινής παρακολούθησης της ατμόσφαιρας κάθε 4 ώρες

Το βασικό λογισμικό υλοποίησης της διαδικασίας αυτής περιγράφεται στην Διπλωματική Εργασία του Χαρβάτη (1990). Όλα τα αποτελέσματα καταγράφονται αυτόματα σε κατάλληλη βάση δεδομένων που περιέχει επίσης πληροφορίες σχετικά με την ημερομηνία και την ώρα της μέτρησης, τη σχετική υγρασία της ατμόσφαιρας και την επικρατούσα θερμοκρασία. Οι μετρήσεις σχετικής υγρασίας και θερμοκρασίας γίνονται μέσω καταλλήλων μεταλλακτών απευθείας συνδεδεμένων με αναλογοψηφιακό μετατροπέα, τοποθετημένο πάνω στο Η/Υ HP-9000/320 (βλ. και την Διπλωματική Εργασία του Μητρόπουλου, 1994).

Κατά τα πρώτα έτη λειτουργίας του συστήματος (1986 – 1988) παρατηρήθηκε ότι υπήρχαν καταγραφές υψηλών μετρήσεων χωρίς να υπάρχει έκλυση ^{137}Cs. Αυτό ήταν ιδιαίτερα εμφανές σε ημέρες βροχής. Μετά από έρευνα της επιστημονικής βιβλιογραφίας, το φαινόμενο εξηγήθηκε από το γεγονός ότι, όπως είναι φυσικό (βλ. π.χ. Fujitaka, 1992), η βροχή συμπαρασύρει προς το έδαφος όλα τα αιωρούμενα στερεά θυγατρικά του ^{222}Rn, ένα από τα οποία είναι και το ^{214}Bi, που εκπέμπει φωτόνια ενέργειας 609 keV. Δεδομένου ότι ο ανιχνευτής NaI που χρησιμοποιείται βρίσκεται κοντά στο έδαφος, είναι φυσικό να αντιλαμβάνεται αυτές τις αυξήσεις της συγκέντρωσης του ^{214}Bi, ως αυξήσεις της συγκέντρωσης ^{137}Cs, αφού τα δύο ισότοπα εκπέμπουν φωτόνια που απέχουν μόλις ~50 keV, ενώ η διακριτική ικανότητα του ανιχνευτή NaI είναι στη γειτονιά των 40 – 60 keV. Παρόλαυτά, υπήρξαν και πολλές

περιπτώσεις όπου διαπιστώθηκαν καταγραφές υψηλών μετρήσεων χωρίς να υπάρχει έκλυση ^{137}Cs και χωρίς να υπάρχει επίσης βροχή, ή άλλη αιτία (όπως π.χ. η πρωινή υγρασία) κατακρήμνισης των θυγατρικών του Ραδονίου προς το έδαφος. Η συστηματική παρακολούθηση αυτών των περιπτώσεων για 15 περίπου χρόνια (1986 – 2001) έδειξε ότι αυτές οι κατά τα άλλα αδικαιολόγητες αυξημένες τιμές με διάρκεια από μία ώρα έως περίπου 2 ημέρες, συνδέονται με σεισμική δραστηριότητα, η οποία κατά μέσο όρο εμφανίζεται 6 έως 10 ημέρες αργότερα, σε μία ακτίνα έως 300 km και με ενεργειακή έκλυση M >≈ 4 (βλ. και Karangelos, 2005). Μετά το εύρημα αυτό το σύστημα αναβαθμίσθηκε ως εξής: προστέθηκε ένας επιπλέον διαχωριστής και σε Daisy Chain, με τον ήδη υπάρχοντα καταμετρητή Model 1776 και τη μονάδα επικοινωνίας Model 2088, ένας επιπλέον καταμετρητής Model 1776 (βλ. και το σχήμα του Διαγράμματος 2.13, με i=2). Ο νέος διαχωριστής ρυθμίστηκε έτσι ώστε το διάστημα τάσεων από σήματα εισόδου, για το οποίο παράγονται λογικοί παλμοί (LL, LL+ΔV) να είναι μέγιστο. Αυτό σημαίνει ότι το LL ρυθμίστηκε στο κατώτερο δυνατό όριο ώστε να μην λογίζεται ως σήμα τάσης ο ηλεκτρονικός θόρυβος και ότι το ΔV ρυθμίσθηκε με τη μέγιστη τιμή του ώστε LL+ΔV = 10 V. Οι ρυθμίσεις αυτές εξυπηρετούν ώστε στον επιπλέον καταμετρητή να καταγράφονται μετρήσεις που αφορούν το σύνολο της ακτινοβολίας που προσβάλλει τον ανιχνευτή και που οφείλεται στην κοσμική ακτινοβολία ή σε άλλες αιτίες. Αν το σύνολο αυτό είναι για κάποιο λόγο αυξημένο, τότε και ο καταμετρητής που παρακολουθεί το ενεργειακό παράθυρο του ^{137}Cs (ή ισοδύναμα του ^{214}Bi), θα καταγράφει και αυτός αυξημένες ενδείξεις, που πιθανότατα δεν οφείλονται σε επερχόμενο σεισμό ή έκλυση ^{137}Cs στην ατμόσφαιρα. Το σύστημα που περιγράφθηκε, υπάρχει συγκροτημένο και ενεργό στο ΕΠΤ-ΕΜΠ, διασφαλίζοντας και την ετοιμότητα του Εργαστηρίου σε σχέση με πιθανό ατύχημα έκλυσης ραδιοϊσοτόπων στον Ελληνικό χώρο.

2.6 Συμπεράσματα - Σχόλια

Στο Κεφάλαιο αυτό περιγράφθηκε συνοπτικά πως χρησιμοποιούνται Πυρηνικά Μετρητικά Συστήματα με Διαχωριστές – Καταμετρητές στο ΕΠΤ-ΕΜΠ. Όπως διαπιστώνεται οι εφαρμογές αυτών των συστημάτων είναι παραγωγικές και λειτουργούν αυτοματοποιημένα, εξαιρετικά οικονομικά και με αξιοπιστία. Εμπλέκουν πολύ απλό εξοπλισμό και δεν χρειάζονται πολύπλοκες μονάδες αναλογοψηφιακής μετατροπής και πολυκαναλικών αναλυτών για την καταγραφή των ζητούμενων συγκεντρώσεων ισοτόπων σε δείγματα εδάφους ή την ατμόσφαιρα.

Επιπλέον αυτά τα συστήματα όπως υπάρχουν σήμερα και λειτουργούν συνεπικουρούν και συμβάλλουν στην θωράκιση της χώρας από ραδιολογικό ατύχημα. Τα συστήματα αυτά όμως απαιτούν περιοδική συντήρηση και αναβάθμιση, κυρίως διότι συστηματικά απαξιώνονται τα υπολογιστικά συστήματα και τα αντίστοιχα λειτουργικά συστήματα, που υποστηρίζουν την συλλογή δεδομένων, λόγω της συνεχούς εξέλιξης σε αυτόν τον τομέα. Χαρακτηριστικά αναφέρεται ότι την δεκαετία του '80, τα συστήματα αυτά ξεκίνησαν να λειτουργούν με τους Η/Υ DEC τύπου PDP 11/04 και LSI-11 (Σιμόπουλος, 1986), στη συνέχεια αναπτύχθηκε η σύνδεσή τους με Η/Υ τύπου PC συμβατού σειράς AT (Βοσυνιώτης, 1990) και τέλος σχεδόν παράλληλα αναπτύχθηκε και η σύνδεσή τους με Η/Υ HP-9000/320-370 (Χαρβάτης, 1990). Στην πορεία αλλαγής του υλικού, απαιτήθηκε και η αντίστοιχη προσαρμογή του λογισμικού στα εξής λειτουργικά συστήματα, αντίστοιχα: το RT-11 της DEC, το DOS της Microsoft και το UNIX System V της Hewlett Packard (το γνωστό HP-UX). Στο επόμενο 3^{ο} Κεφάλαιο, διαπιστώνεται η σημερινή (2008) επιτακτική αναγκαιότητα και περιγράφονται τα αναγκαία βήματα, για μία επιπλέον αναβάθμιση των συστημάτων αυτών, τόσο σε επίπεδο υλικού, όσο και σε επίπεδο λογισμικού κάτω από το λειτουργικό σύστημα LINUX. Για το σκοπό αυτό προτείνονται τόσο οι κατάλληλες εκδόσεις λογισμικού, όσο και υλικό πρόσφορο για τις απαραίτητες δοκιμές.

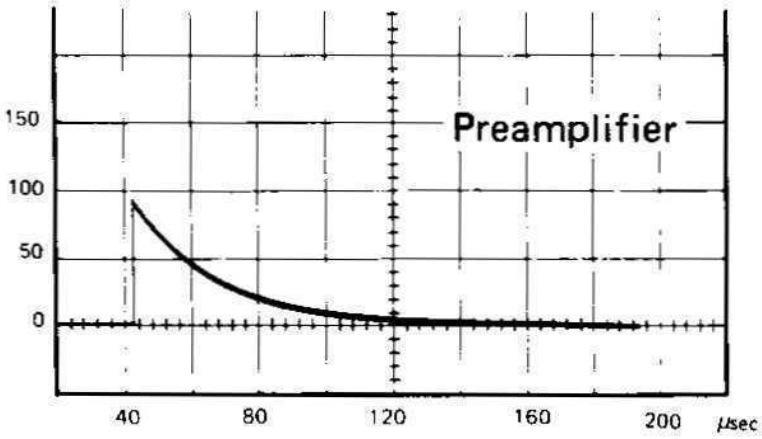
ΔΙΑΓΡΑΜΜΑΤΑ ΤΟΥ 2^{ου} ΚΕΦΑΛΑΙΟΥ

Διάγραμμα 2.1

Τυποποιημένα ηλεκτροστατικά κουτιά NIM

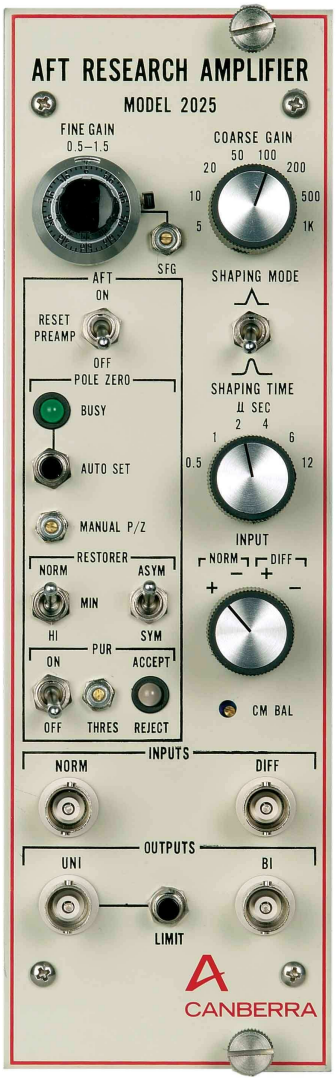
Διάγραμμα 2.2

Προενισχυτής NIM, εμπρός όψη (πάνω), πίσω όψη (κάτω)



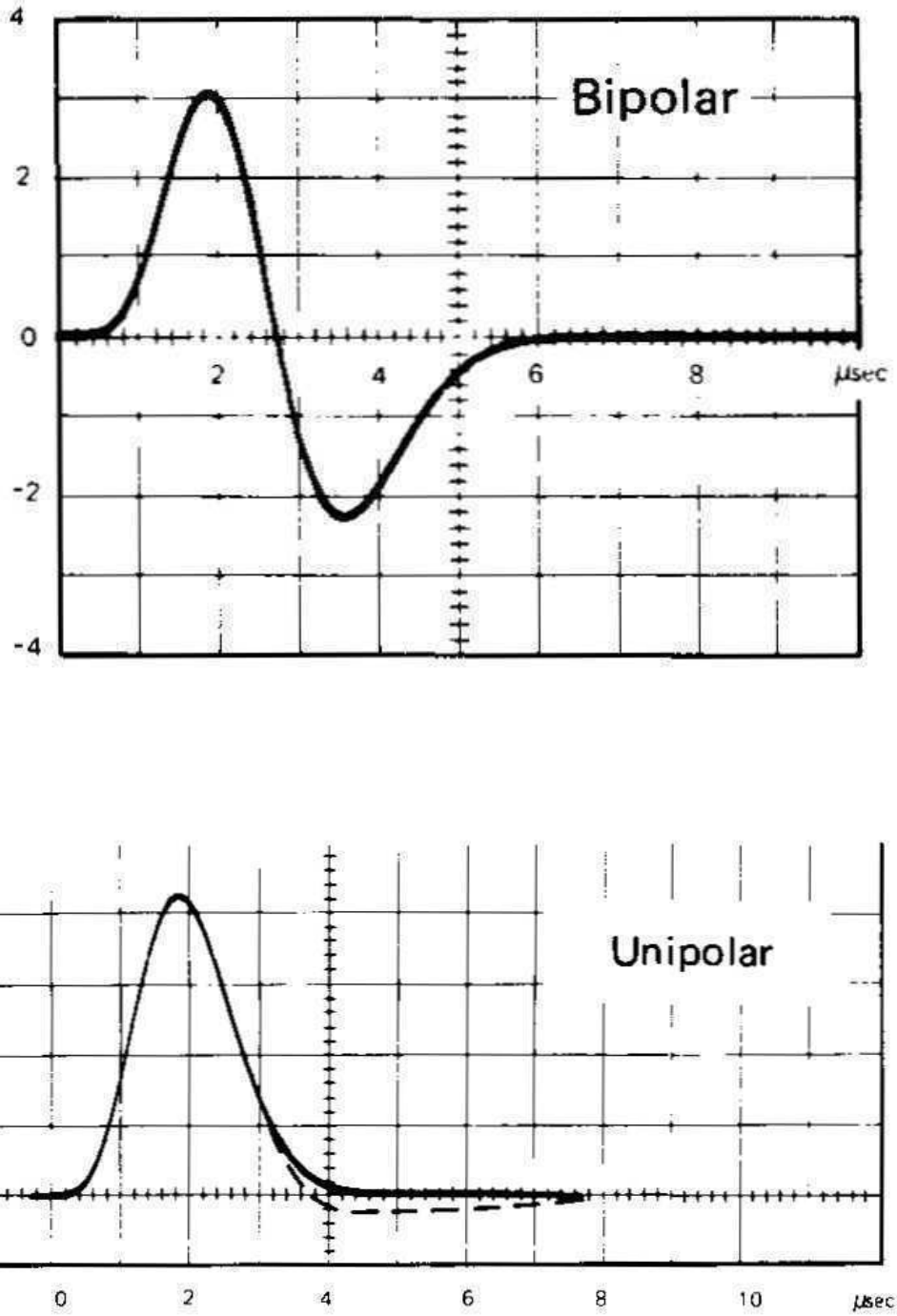
Διάγραμμα 2.3

Τυπική μορφή σήματος τάσεως στην έξοδο προενισχυτή



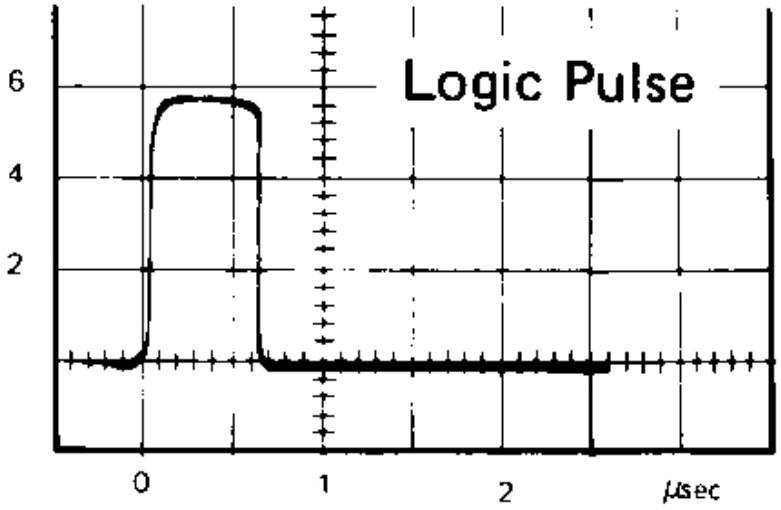
Διάγραμμα 2.4

Όψη ενισχυτή NIM (CANBERRA, Model 2025)



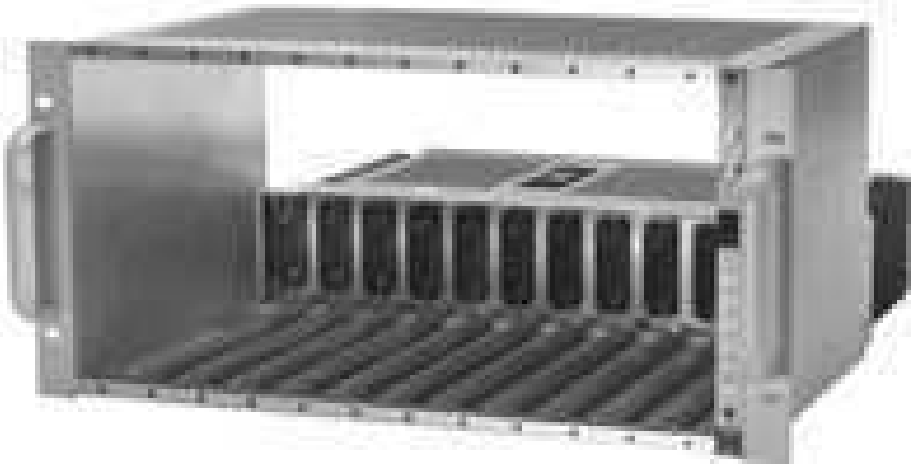
Διάγραμμα 2.5

Τυπική μορφή σήματος τάσεως στην έξοδο ενισχυτή,
Bipolar (πάνω), Unipolar (κάτω)



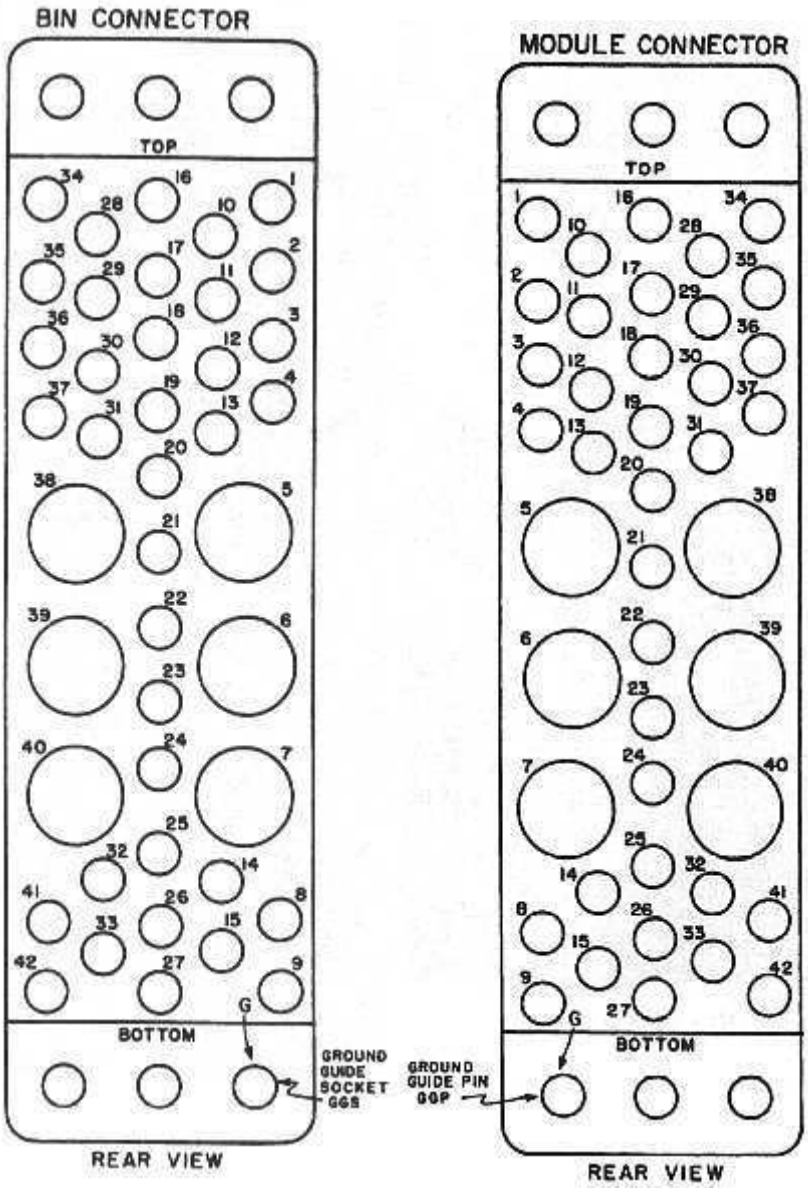
Διάγραμμα 2.6

Τυπική μορφή λογικού παλμού τάσεως



Διάγραμμα 2.7

NIM Bin (CANBERRA, Model 2000 / 2100)



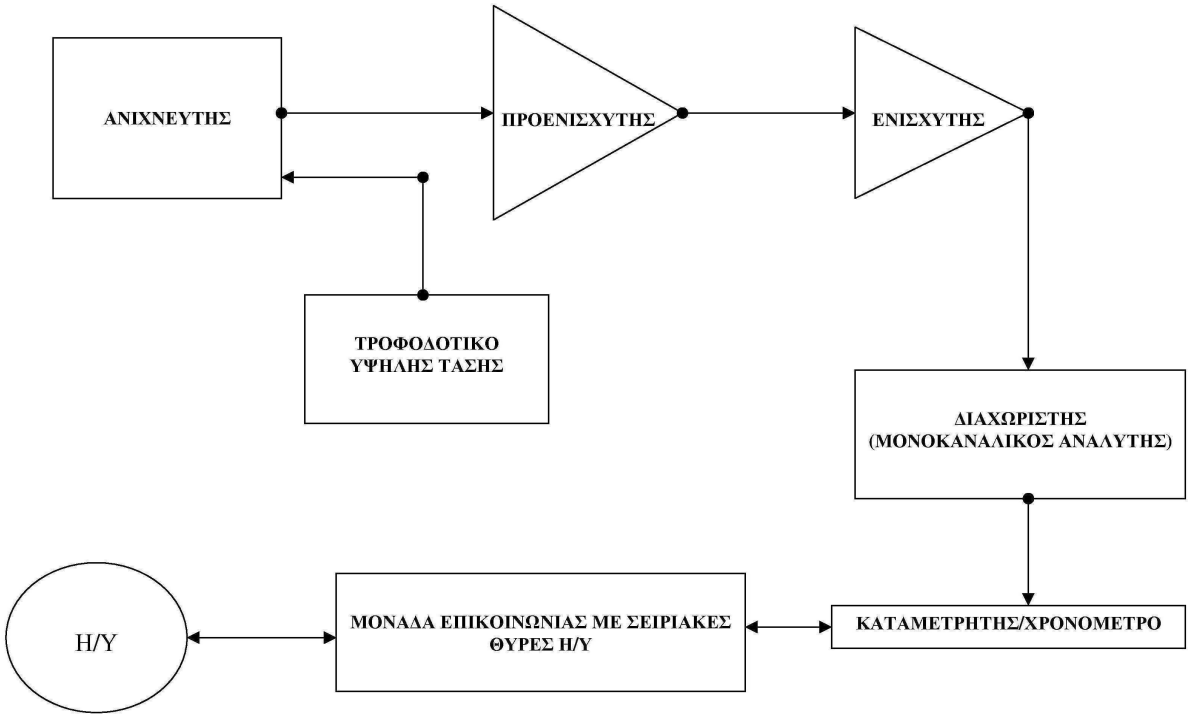
Διάγραμμα 2.8

NIM Connectors,
Θηλυκό (αριστερά – επί της NIM Bin), Αρσενικό (δεξιά – επί της μονάδας NIM)

PIN	FUNCTION
1	RESERVED
2	RESERVED
3	**BIN GATE**
4	RESERVED
5	
6	
7	
8	+200 V D.C.
9	SPARE
* 10	+6 V
* 11	-6 V
12	RESERVED
13	SPARE
14	SPARE
15	RESERVED
* 16	+12 V
* 17	-12 V
18	SPARE
19	RESERVED
20	SPARE
21	SPARE
22	RESERVED
23	RESERVED
24	RESERVED
25	RESERVED
26	SPARE
27	SPARE
* 28	+24 V
* 29	-24 V
30	SPARE
31	SPARE
32	SPARE
33	117 V A.C. (HOT)
* 34	POWER RETURN GND
* 35	RESET
36	GATE
37	SPARE
38	
39	
40	
* 41	117 V A.C. (NEUTRAL)
* 42	HIGH QUALITY GND
G	GROUND GUIDE PIN

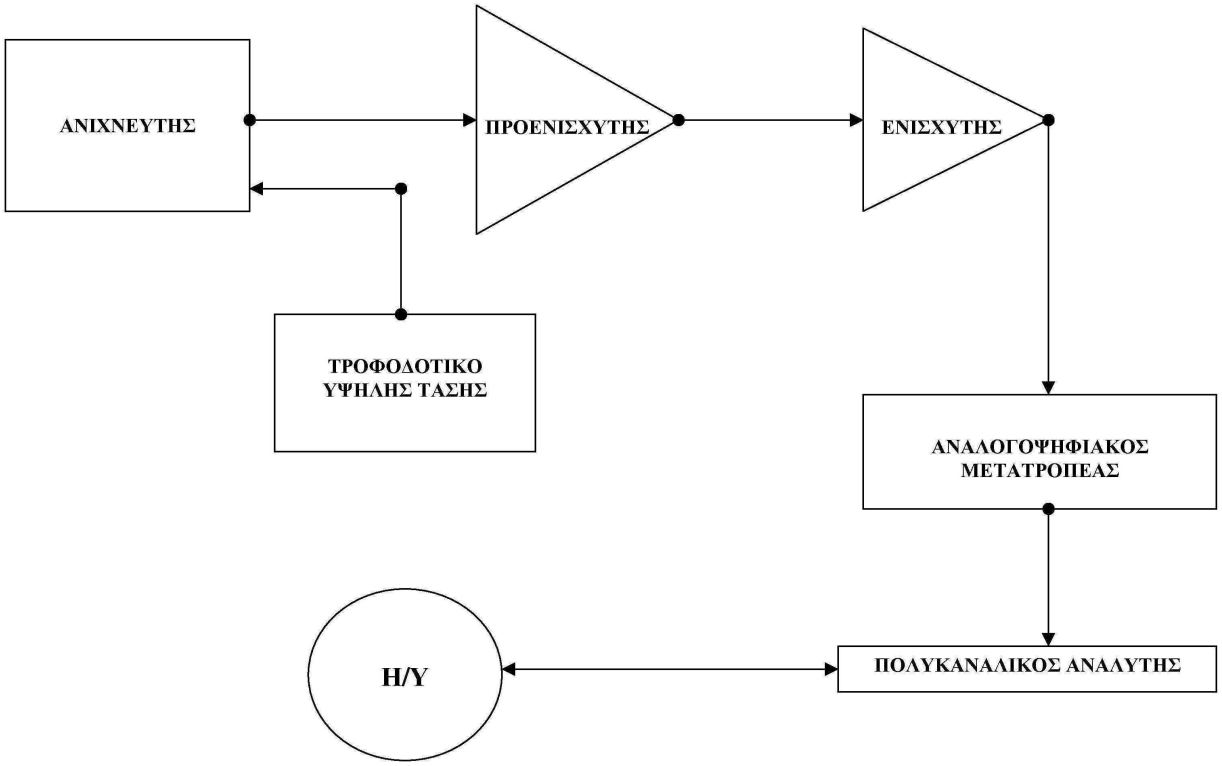
Διάγραμμα 2.9

Τυποποίηση NIM Connector



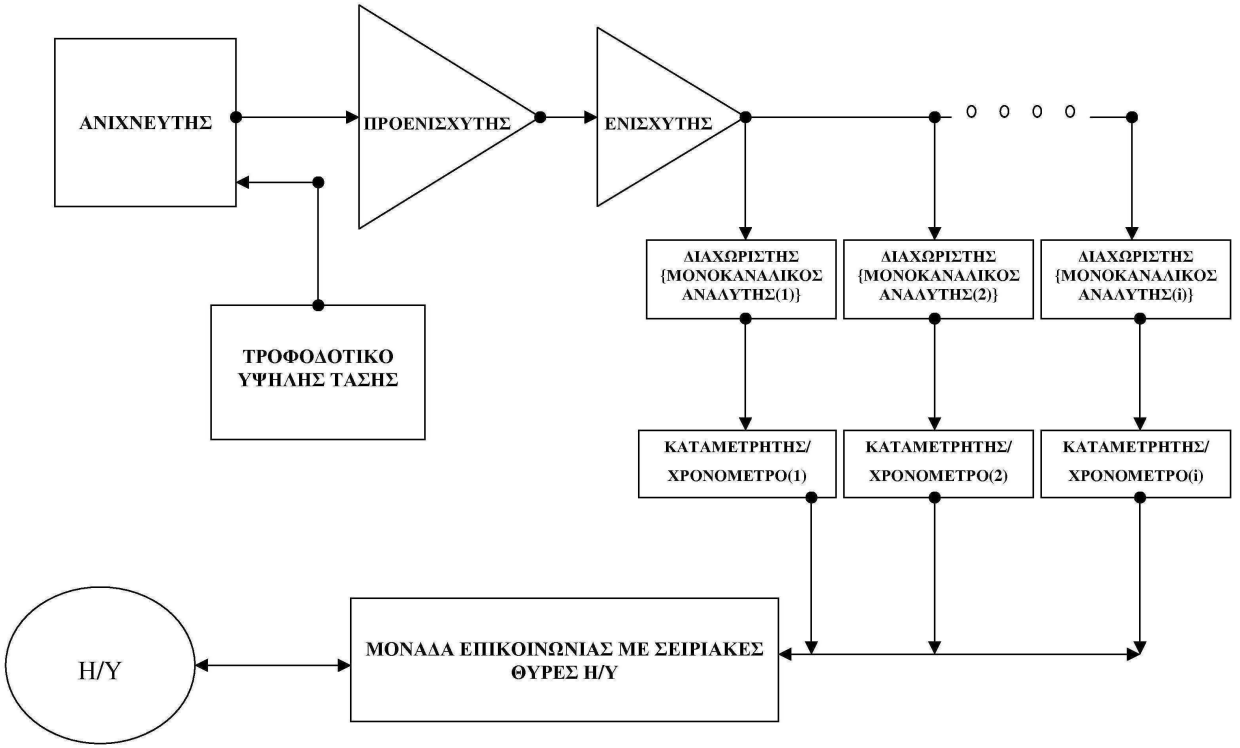
Διάγραμμα 2.10

Απλή διάταξη με μονάδες NIM (με διαχωριστή και καταμετρητή)



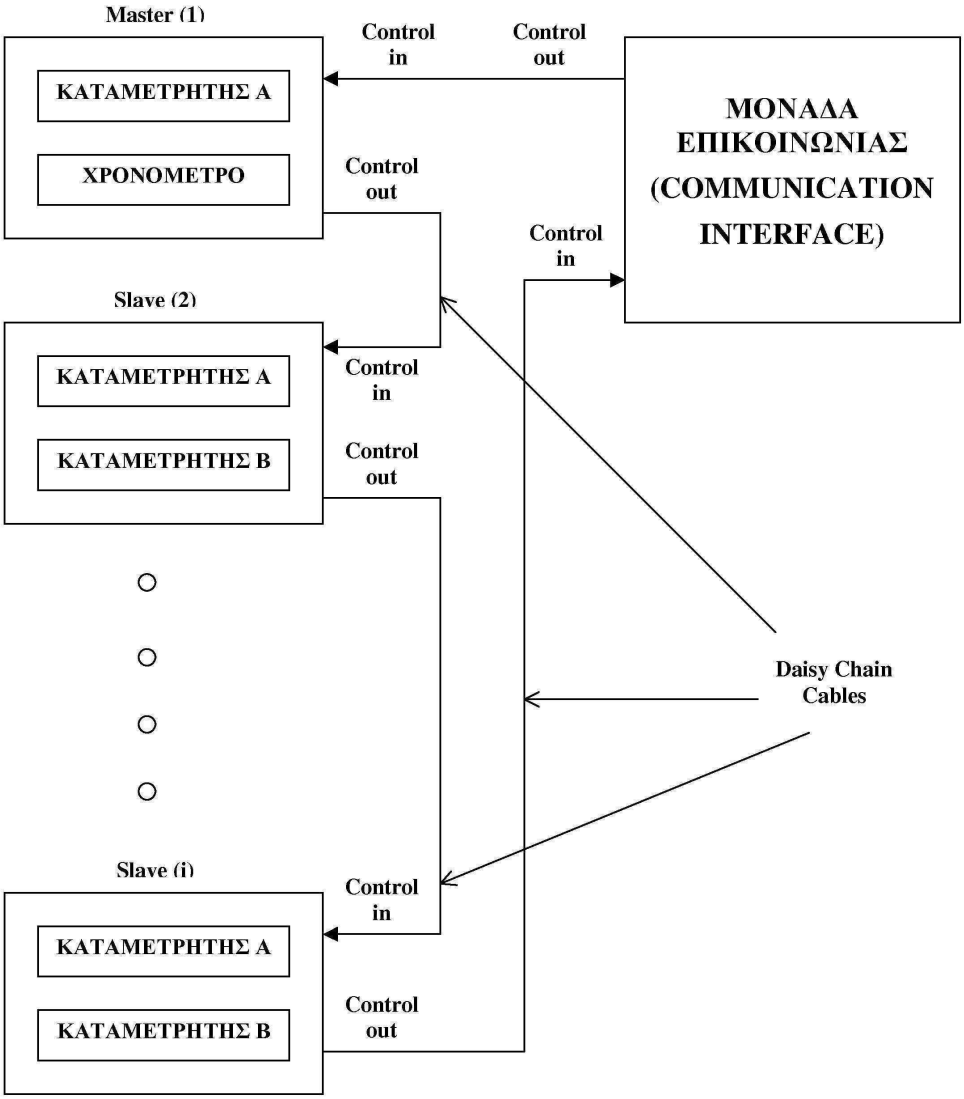
Διάγραμμα 2.11

Απλή διάταξη με μονάδες NIM (με αναλογοψηφιακό μετατροπέα και πολυκαναλικό αναλυτή)



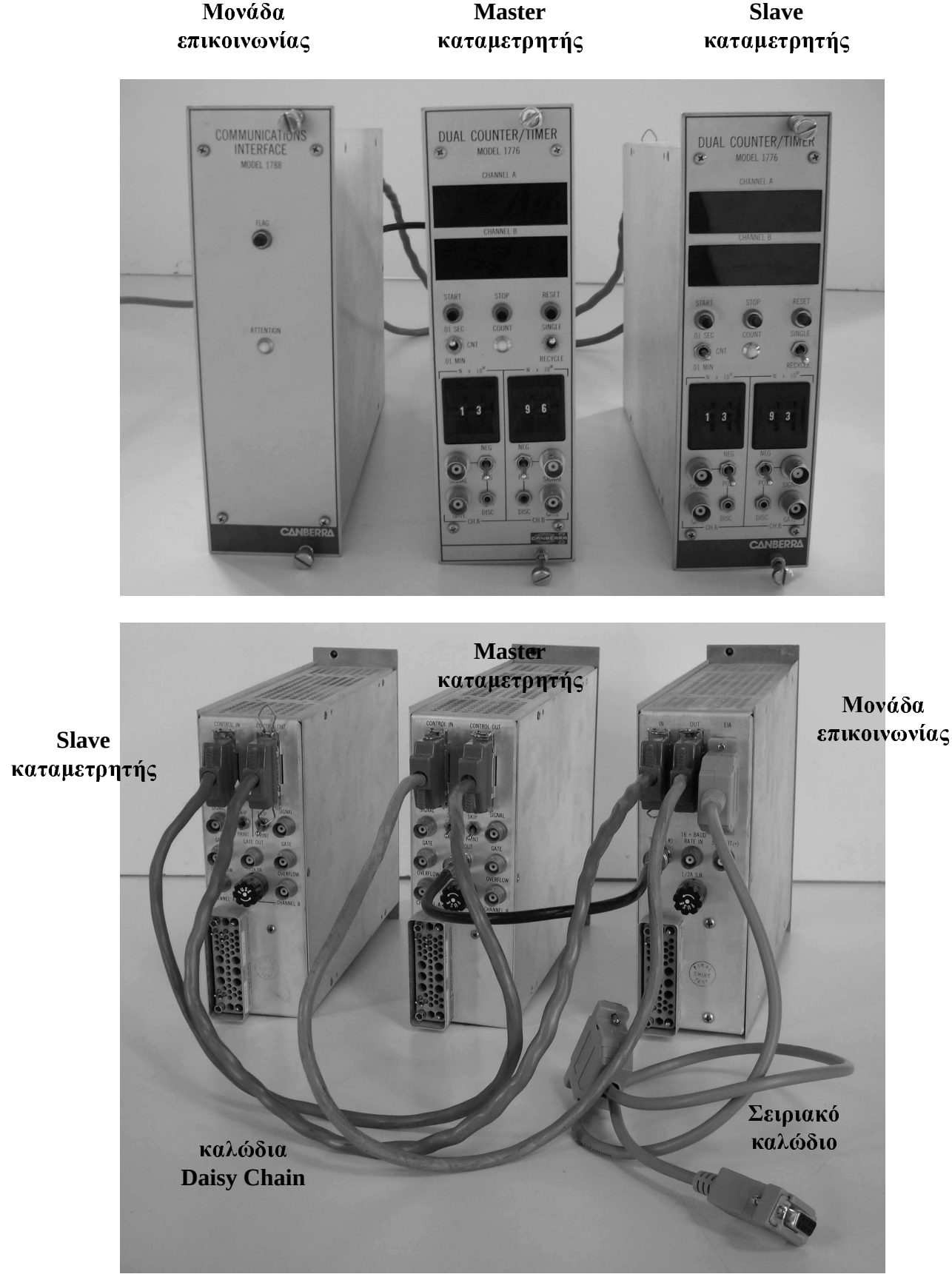
Διάγραμμα 2.12

Διάταξη με μονάδες NIM (πολλά ζεύγη διαχωριστών - καταμετρητών)



Διάγραμμα 2.13

Σχηματική αναπαράσταση διάταξης Daisy Chain με Dual Counter/Timers (Ο χρόνος ρυθμίζεται από το χρονόμετρο του καταμετρητή Master)

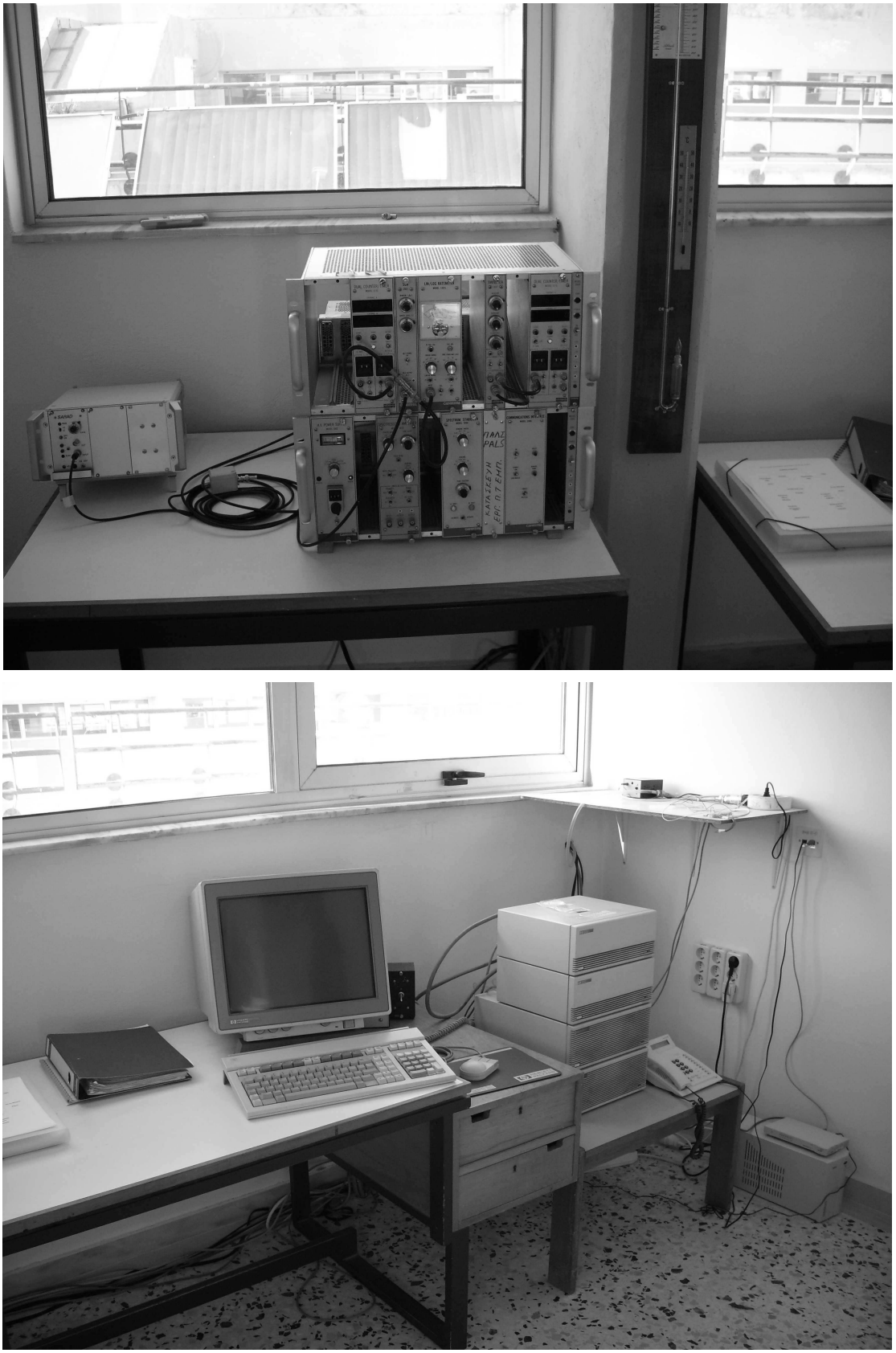


Διάγραμμα 2.14

Αναπαράσταση διάταξης Daisy Chain (μονάδες εκτός NIM Bin): Πρόοψη (πάνω), Πίσω όψη (κάτω), με δύο καταμετρητές Dual Counter/Timer (Model 1776) και μονάδα επικοινωνίας (Model 1788)

Διάγραμμα 2.15

Διάταξη μέτρησης συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI (Πυρηνικά Ηλεκτρονικά – Ανιχνευτής – Η/Υ)



Διάγραμμα 2.16

Διάταξη μέτρησης συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή, Πυρηνικά Ηλεκτρονικά (πάνω), Η/Υ (κάτω)

ΚΕΦΑΛΑΙΟ 3
ΣΥΓΚΡΟΤΗΣΗ ΑΠΑΡΑΙΤΗΤΟΥ ΥΛΙΚΟΥ, ΕΓΚΑΤΑΣΤΑΣΗ Λ/Σ LINUX ΚΑΙ ΒΑΣΙΚΟΥ ΛΟΓΙΣΜΙΚΟΥ

3.1 Εισαγωγή - Ιστορικό

Το ΕΠΤ- ΕΜΠ διαθέτει, όπως αναφέρθηκε και στο προηγούμενο 2^{ο} Κεφάλαιο, μία σειρά από συσκευές που εξυπηρετούν Πυρηνικά Μετρητικά Συστήματα, μερικές από τις οποίες είναι δυνατόν να χρησιμοποιούνται ώστε οι μετρήσεις από αυτά να μεταφέρονται αυτόματα σε Η/Υ μέσω σειριακής επικοινωνίας. Οι υπόψιν συσκευές (κυρίως καταμετρητές) ήταν συνδεδεμένες μέχρι και το 1987 σε συστήματα Η/Υ DEC τύπου PDP-11/04 και τύπου LSI-11 (Σιμόπουλος, 1986). Το 1987 στα πλαίσια εκσυγχρονισμού του ΕΠΤ-ΕΜΠ, αποκτήθηκαν δύο Η/Υ HP-9000/320 (επεξεργαστής Motorola MC68000, Λ/Σ UNIX) και ένας HP-VECTRA (επεξεργαστής Intel 80286, 8MHz, Λ/Σ DOS). Οι υπολογιστές αυτοί σύντομα (1987 – 1989) διασυνδέθηκαν μεταξύ τους καθώς και με τον κεντρικό τότε Η/Υ του ΕΜΠ CDC-CYBER 171. Περί το 1990 αποκτήθηκε ένας επιπλέον Η/Υ HP-9000/370 (επεξεργαστής Motorola 68000, Λ/Σ UNIX).

Από τους πιο πάνω Η/Υ του ΕΠΤ-ΕΜΠ οι δύο HP-9000/320-370 ήταν εκείνοι που είχαν τις μεγαλύτερες δυνατότητες σε επεξεργασία και αποθήκευση δεδομένων. Κατά συνέπεια επωμίσθηκαν σχεδόν όλα τα καθήκοντα συλλογής, αποθήκευσης και επεξεργασίας δεδομένων του ΕΠΤ-ΕΜΠ. Ειδικό λογισμικό αναπτύχθηκε (Χαρβάτης, 1990) έτσι ώστε οι δύο Η/Υ HP-9000/320-370 να μπορούν να συνδεθούν με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας

Ο HP-VECTRA αποκτήθηκε με σκοπό να είναι ένα εφεδρικό μηχάνημα για τα συστήματα HP-9000. Ανήκει στην κατηγορία των Η/Υ τύπου PC συμβατός (IBM compatibles) που είναι πολύ πλατειά διαδεδομένοι σ'όλο τον κόσμο καθώς και στην Ελλάδα. Το HP-VECTRA ήταν λοιπόν ένα μηχάνημα το οποίο μπορούσε να αντικατασταθεί πολύ εύκολα. Ο λόγος είναι απλός: το λογισμικό που έχει γραφτεί γι'αυτό, μπορεί να εκτελεστεί από οποιοδήποτε μηχάνημα ανήκει στην κατηγορία των Η/Υ τύπου PC συμβατού. Ειδικό λογισμικό αναπτύχθηκε (Βοσυνιώτης, 1990) έτσι ώστε ως εφεδρικό σύστημα Η/Υ του ΕΠΤ-ΕΜΠ ο HP-VECTRA να μπορεί να συνδεθεί με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας, για να

μπορούσε σε περίπτωση ανάγκης να αναλάβει τις εργασίες που επιτελούσαν τα δύο συστήματα HP-9000/320-370.

Οι δύο Η/Υ HP-9000/320-370 έφεραν σε πέρας τα καθήκοντα τους με απόλυτη επιτυχία μέχρι και τον Ιανουάριο 2008. Κατά τη διάρκεια της ευδόκιμης ζωής τους τα δύο αυτά συστήματα, συντηρούνταν συστηματικά το διάστημα από 1987 έως και 1997, τόσο σε επίπεδο υλικού, όσο και σε επίπεδο λογισμικού. Από το 1998 και μετά δεν υπήρχε πλέον δυνατότητα συντήρησης και αναβάθμισής τους λόγω απόσυρσης των προϊόντων αυτών από την μητρική εταιρεία Hewlett Packard. Επιπλέον το έτος 2000 (τη στιγμή 31 Δεκεμβρίου 1999 – 1 Ιανουαρίου 2000), αυτοί οι Η/Υ αντιμετώπισαν και σημαντικά προβλήματα "έτους 2000" (στα αγγλικά "Y2K problems"), τα οποία λύθηκαν μεν επιτυχώς αλλά εκ των ενόντων και χωρίς υποστήριξη του κατασκευαστή. Για αυτούς τους Η/Υ, από ένα σημείο και μετά, οι αναγκαίες αντικαταστάσεις καταστραμμένου υλικού ήταν δυνατές μόνο μέσω της παράλληλης αγοράς μεταχειρισμένων ανταλλακτικών. Περί το 2005 πλέον και αυτή η αγορά έγινε ασύμφορη και άρχισε να εξετάζεται και να σχεδιάζεται η μετάβαση σε νέα συστήματα κάτω από το Λ/Σ LINUX, οπότε ο ένας Η/Υ HP-9000/370 αντικαταστάθηκε (2008) από νέο σύστημα κάτω από το Λ/Σ LINUX. Παρόλαυτά, δεν αναπτύχθηκε το ειδικό λογισμικό που απαιτείται ώστε το υλικό με το Λ/Σ LINUX να μπορεί να συνδεθεί με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας. Έτσι:

(α) το τμήμα καθηκόντων που αφορά "μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI" εξακολουθεί και σήμερα (2008) να εξυπηρετείται από τον ακόμα λειτουργικό Η/Υ HP-VECTRA. Σημειώνεται όμως ότι αυτό δεν είναι σημαντικό πρόβλημα παρόλη την απαξίωση του υλικού HP-VECTRA (ήδη από το 1990 – 1992), διότι το λογισμικό που αναπτύχθηκε για αυτόν, μπορεί να λειτουργεί ακόμα με τον ίδιο αποτελεσματικό τρόπο, σε οποιοδήποτε PC συμβατό, οποιασδήποτε σημερινής (2008) και ελπίζεται και μελλοντικής τεχνολογίας. Ο λόγος είναι ότι, οι νεότεροι επεξεργαστές των PC συμβατών, βασίζονται στην ίδια αρχιτεκτονική με τους πρωτότυπους Intel 80x86.

(β) το τμήμα καθηκόντων που αφορά στις "μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI" εξακολουθεί και σήμερα (2008) να εξυπηρετείται από τον ακόμα λειτουργικό Η/Υ HP-9000/320. Μάλιστα οι σχετικές μετρήσεις καταχωρούνται και στην κύρια σελίδα του δικτυακού τόπου του

Τομέα Πυρηνικής Τεχνολογίας (http://nuclear.ntua.gr) με τεχνικές που αναπτύχθηκαν από τους Χίνη, Σιμόπουλο (1995). Απαιτείται όμως να αναβαθμισθεί το υπάρχον ειδικό λογισμικό σειριακών επικοινωνιών ώστε το υλικό με το Λ/Σ LINUX να μπορεί να συνδεθεί με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας. Η μετατροπή του υπόψιν λογισμικού είναι, όπως έχει σημειωθεί το αντικείμενο της παρούσης Διπλωματικής Εργασίας.

3.2 Το Λ/Σ UNIX και η εξέλιξή του στο Λ/Σ LINUX

Τα δύο συστήματα Η/Υ HP-9000/320-370 διέθεταν το λειτουργικό σύστημα HP-UX της Hewlett Packard. Το λειτουργικό σύστημα HP-UX ανήκει στην ευρύτερη οικογένεια συστημάτων UNIX System V. Το UNIX πρωτοεμφανίστηκε στις αρχές της δεκαετίας του '70 και από τότε η εξάπλωσή του ήταν ραγδαία. Δύο είναι τα καθοριστικά στοιχεία που οδήγησαν στην μεγάλη επιτυχία του UNIX:

1. Το μικρό μέγεθος και η λιτότητα που χαρακτηρίζει τον πυρήνα του (kernel) που είναι σήμερα, ο μικρότερος σε μέγεθος πυρήνας λειτουργικού συστήματος time-sharing.

2. Το ότι είναι σε μεγάλο βαθμό ανεξάρτητο από το υλικό του υπολογιστή. Για να μεταφερθεί δηλαδή το UNIX σε ένα νέο υπολογιστή μόνο ένα μικρό τμήμα του πυρήνα (kernel) του συστήματος χρειάζεται να γραφτεί από την αρχή, ενώ για το μεγαλύτερο τμήμα (που είναι γραμμένο σε γλώσσα C) αρκεί μια νέα μετάφραση (compilation) με τον compiler του νέου υπολογιστή. Το στοιχείο αυτό, που ονομάζεται φορητότητα (στα αγγλικά portability), ανακλάται και στις εφαρμογές που γράφονται κάτω από αυτό το λειτουργικό σύστημα.

Η εξέλιξη του λειτουργικού συστήματος UNIX παρουσιάζεται στο σχήμα του Διαγράμματος 3.1. Μέχρι και το 1981, το Λ/Σ UNIX ήταν περιορισμένης χρήσης σε συστήματα "main frame" μέσα σε πανεπιστημιακά τμήματα υπολογιστικών γνωστικών αντικειμένων και σε ερευνητικά εργαστήρια συνδεδεμένα με το Υπουργείο Αμύνης των Η.Π.Α. καθώς και σε διάφορα τμήματα έρευνας και ανάπτυξης σε βιομηχανίες και οργανισμούς. Την εποχή εκείνη άρχισε η επέκταση του UNIX σε μηχανές μικρότερων δυνατοτήτων, τους λεγόμενους Η/Υ "super mini", που βασίζονταν στους επεξεργαστές της σειράς MC68000 της Motorola. Αυτό επέτρεψε τη διάδοση του UNIX σε εφαρμογές μικρής και μεσαίας κλίμακας, όπως οι εφαρμογές γραφείου ή ακόμα και οι οικιακές εφαρμογές και την παγίωση του

προτύπου System V (1984) μέχρι την εμφάνιση του Λ/Σ Linux. Η διάδοση αυτή διάρκεσε κατά προσέγγιση μέχρι και περίπου το 1995. Η διάθεση του λειτουργικού συστήματος γινόταν συνήθως μαζί με το υλικό (δηλ. τον Η/Υ) από επώνυμες εταιρείες κατασκευής, όπως η Hewlett Packard, η Sun Microsystems, η Silicon Graphics Inc. και άλλες. Κάθε μία από αυτές τις εταιρείες υποστήριζε τη δική της έκδοση UNIX, για την οποία διατηρούσε πνευματικά δικαιώματα.: π.χ. η Hewlett Packard για το HP-UX και επεξεργαστές Motorola MC68000, η Sun για το Solaris και επεξεργαστές Motorola MC68000 ή SPARC, και η Silicon Graphics για το IRIX και επεξεργαστές Toshiba R4000.

Παράλληλα με τη διάδοση του UNIX System V, άρχισε (1983 – 1985) από τον Αμερικανό Richard Stallman και την ομάδα του η ανάπτυξη του λειτουργικού συστήματος GNU, το οποίο βασίζονταν στην φιλοσοφία του "ελεύθερου λογισμικού", δηλ. του λογισμικού που μπορεί να χρησιμοποιείται χωρίς κόστος, άδεια και άλλους περιορισμούς, και στις αρχές προγραμματισμού στις οποίες βασίσθηκε το πρωτότυπο UNIX, χωρίς όμως να χρησιμοποιεί έτοιμα τμήματα και βιβλιοθήκες από αυτό το Λ/Σ. Ο στόχος του Stallman ήταν η δημιουργία ενός Λ/Σ ελεύθερου και συμβατού με το Λ/Σ UNIX. Το 1985 ο Stallman δημιούργησε το Ίδρυμα Ελεύθερου Λογισμικού (στα αγγλικά Free Software Foundation) και διατύπωσε την ιδέα για το λογισμικό ελεύθερης άδειας γνωστή ως GNU GPL (GNU General Public License). Περί το 1987, με το ίδιο σκεπτικό, και για εκπαιδευτικούς σκοπούς, ξεκίνησε η δημιουργία του Λ/Σ Minix, από τον Ολλανδός Andrew S. Tanenbaum, το οποίο λογίζεται ως κλώνος του Λ/Σ Unix, που είχε την δυνατότητα συνεργασίας με υλικό βασισμένο στους περισσότερους τότε διαθέσιμους επεξεργαστές.

Στα πλαίσια του Λ/Σ GNU και περί τα μέσα του 1990 είχαν αναπτυχθεί αρκετά προγράμματα που ήταν απαραίτητα για αυτό το Λ/Σ π.χ. διαμορφωτές κειμένου, μεταφραστές, παραθυρικό περιβάλλον κ.α., παρόλο που πολλά ακόμα για την επικοινωνία σε χαμηλότερο επίπεδο π.χ. προγράμματα εγκατάστασης συσκευών, πυρήνας Λ/Σ είχαν καθυστερήσει και ήταν ημιτελή. Αυτός ήταν ο λόγος που έκανε τον Φιλανδό Linus Torvalds να αναπτύξει το 1991, το δικό του ελεύθερο Λ/Σ. Το αρχικό πρότυπο του Torvalds ήταν το Minix, αλλά ο Tannenbaum δεν επέτρεψε την περαιτέρω ανάπτυξή του από άλλους. Για το λόγο αυτό, ο Torvalds δημιούργησε εξαρχής έναν πυρήνα για το αντικαταστήσει. Αρχικά ο πυρήνας αυτός ονομάστηκε

FreaX [free-freak-uni(X)] αλλά αργότερα έλαβε την ονομασία LINUX, έναν όρο που επινόησε ο Ari Lemmke.

Δεδομένου ότι το LINUX βασίσθηκε σε συσσωρευμένη εμπειρία και εστίασε σε πυρήνα που συνεργάζεται με τους πολύ διαδεδομένους επεξεργαστές Intel 80x86 και το υπόλοιπο υλικό των Η/Υ που είναι PC συμβατοί, γνώρισε μεγάλη επιτυχία και είναι σήμερα (2008) το κυρίαρχο Λ/Σ σε επαγγελματικές / ερευνητικές εφαρμογές. Οι περισσότερες εφαρμογές και βιβλιοθήκες του προέρχονταν από το Λ/Σ GNU, αλλά παρόλαυτά το LINUX θεωρείται κλώνος του Λ/Σ UNIX, αφού διαθέτει τις περισσότερες εντολές του, ενώ η φιλοσοφία της σχεδίασής του πλησιάζει περισσότερο το UNIX από οποιοδήποτε άλλα λειτουργικά συστήματα.

Το LINUX αναπτύχθηκε με βάση το πρότυπο POSIX (Portable Operating System-IX) που είναι το όνομα μιας σειράς πρότυπων που δημιουργήθηκαν το 1988, τα οποία καθορίζουν τις βασικές κλήσεις ενός λειτουργικού συστήματος και πιο συγκεκριμένα το API (Application Programming Interface). Το POSIX δημιουργήθηκε με αφορμή την προσπάθεια του Stallman για το GNU, με στόχο την συμβατότητα των Λ/Σ UNIX έτσι ώστε οι κώδικες που τρέχουν σε έναν υπολογιστή με Λ/Σ UNIX να μπορούν να τρέχουν σε οποιοδήποτε άλλο υπολογιστή με Λ/Σ UNIX χωρίς να χρειάζεται να αλλάξει ο κώδικας (ή μέρος αυτού).

Ως αποτέλεσμα της επιτυχίας του LINUX και της ελεύθερης διάδοσης του κώδικα στον οποίο βασίζεται, διατίθενται σήμερα πολλές εκδόσεις αυτού του Λ/Σ. Σχετικά αναφέρονται οι ακόλουθες γνωστότερες σε κατάταξη που περίπου αντικατοπτρίζει την δημοφιλία τους και την ποιότητά τους:

- Red Hat και κλώνοι
- Debian και κλώνοι
- Slackware και κλώνοι
- Mandriva και κλώνοι
- Fedora Core και κλώνοι

3.3 Το UNIX και το LINUX στο ΕΠΤ-ΕΜΠ

Πέρα από τους δύο Η/Υ HP-9000/320-370 με ΛΣ UNIX HP-UX 6.x έως 7.x, οι οποίοι πρωτολειτούργησαν στο ΕΠΤ-ΕΜΠ, τη δεκαετία του 1980, περί το 1995

στους Η/Υ του Εργαστηρίου εντάχθηκε και ένα σύστημα Η/Υ "Indy" της Silicon Graphics Inc. με Λ/Σ UNIX Irix 5.x. Το σύστημα αυτό ανέλαβε κυρίως το φόρτο του e-mail του προσωπικού του ΕΠΤ-ΕΜΠ και επίσης το δικτυακό τόπο του Τομέα Πυρηνικής Τεχνολογίας ΕΜΠ, με τη γνωστή URL: http://nuclear.ntua.gr. Λόγω των υψηλών –για την εποχή του- τεχνικών χαρακτηριστικών του συστήματος αυτού, δηλ. ταχύτητα, μνήμη κ.ο.κ., στον Η/Υ Indy, γίνονταν επίσης χρονοβόρες επεξεργασίες παραγωγής ραδιομετρικών χαρτών σχετικά με την κατανομή φυσικών και τεχνιτών ραδιοϊσοτόπων στα επιφανειακά εδάφη της Ελλάδας και οδήγησαν σε μία σημαντική σειρά δημοσιεύσεων σε επιστημονικά περιοδικά (βλ. π.χ. και Anagnostakis, 1997). Ο Η/Υ Indy, λειτούργησε απρόσκοπτα για το ΕΠΤ-ΕΜΠ επί δεκαετία (μέχρι το 2006). Η αντικατάστασή του κρίθηκε αναγκαία λόγω αδυναμίας συντήρησης του λογισμικού και του υλικού. Στη θέση του τοποθετήθηκε ένας Η/Υ PC συμβατός με Λ/Σ LINUX. Έτσι ο Η/Υ αυτός έγινε ο πρώτος στον οποίο εγκαταστάθηκε Λ/Σ LINUX στο ΕΠΤ-ΕΜΠ. Η έκδοση του Λ/Σ LINUX που προτιμήθηκε, ήταν η Debian (έκδοση Sarge), διότι διαπιστώθηκε ότι αυτή ήταν η πλέον δημοφιλής και διαδεδομένη στο εσωτερικό του ΕΜΠ, χωρίς να προκύπτει κάποιο κόστος. Το γεγονός αυτό επέτρεψε την γρήγορη απάντηση ερωτημάτων σχετικά με την εγκατάσταση, την οργάνωση και την ασφάλεια του Λ/Σ και την αντιμετώπιση τυχόν προβλημάτων.

Σημειώνεται ότι το Λ/Σ Debian, είναι μια δημοφιλής διανομή LINUX, αποτέλεσμα της εργασίας των μελών της κοινότητας Debian Project. Πρόκειται για ελεύθερο λογισμικό που αναπτύσσεται μέσω της συνεργασίας εθελοντών από όλο τον κόσμο. Βασίζεται στον πυρήνα LINUX και στην ομάδα βασικών εργαλείων του Λ/Σ GNU. Το Debian σαν σχέδιο εργασίας άρχισε το 1993 από τον Γερμανό Ian Murdock, φοιτητή τότε του πανεπιστημίου Purdue Η.Π.Α, ο οποίος συνέταξε το λεγόμενο "Μανιφέστο Debian", που καλούσε για τη δημιουργία ενός Λ/Σ LINUX, που θα αναπτύσσονταν με τρόπο ανοιχτό στο πνεύμα του GNU και θα κάλυπτε την αρχιτεκτονική πολλών διαφορετικών επεξεργαστών. Από το 1996 μέχρι σήμερα (2008) έχουν δημιουργηθεί εννέα διαφορετικές εκδόσεις. Στη διάρκεια ζωής του Debian, πολλά άλλα Λ/Σ LINUX το χρησιμοποίησαν σαν βάση για να αναπτυχθούν. Τα πιο γνωστά σήμερα (2008) είναι τα Ubuntu, Xandros και Knoppix.

Αφού διαπιστώθηκε ότι η εγκατάσταση και η λειτουργία του Λ/Σ Debian, υπήρξε ομαλή και μάλλον απρόσκοπτη, και δεδομένου ότι την περίοδο 2005 – 2007 είχε αποδειχθεί μετά από συνεχείς βλάβες, ότι η συνέχεια της λειτουργίας των δύο

συστημάτων Η/Υ HP-9000/320-370 ήταν πολύ δύσκολο να διατηρηθεί λόγω παρωχημένης τεχνολογίας και έλλειψης ανταλλακτικών, αποφασίσθηκε (2007) να αντικατασταθεί τουλάχιστον το ένα από τα δύο συστήματα, το HP-9000/370, από ένα σύστημα PC συμβατό με Λ/Σ LINUX. Η έκδοση του Λ/Σ LINUX που προτιμήθηκε, ήταν η Fedora Core 4 (Stentz), κυρίως διότι

(α) υπήρχε διαθέσιμο λογισμικό εγκατάστασης (drivers) για υλικό (κάρτες PCI) σειριακής επικοινωνίας και αναλογοψηφιακής μετατροπής της εταιρείας ADVANTECH, το οποίο υπάρχει σε αφθονία στο ΕΠΤ-ΕΜΠ. Η εταιρεία ADVANTECH είναι αξιόπιστος και με πολύ καλό λόγο κόστος / απόδοση κατασκευαστής υλικού για τη συλλογή δεδομένων απευθείας από Η/Υ. Η αξιοπιστία και η απόδοση των προϊόντων της υπόψιν εταιρείας έχει πολλαπλώς αποδειχθεί σε σειρά Διπλωματικών Εργασιών που έχουν εκπονηθεί σχετικά στο ΕΠΤ-ΕΜΠ.

(β) ήταν σχεδόν πλήρως συμβατή με την έκδοση του Λ/Σ LINUX Red Hat Enterprise Linux 5. Είναι γνωστό ότι οι εκδόσεις Red Hat του Λ/Σ LINUX, είναι σήμερα οι πλέον ολοκληρωμένες και αποδεδειγμένα αξιόπιστες.

(γ) δεν είχε κόστος.

Σημειώνεται ότι το Λ/Σ Fedora Core είναι ένα γενικής χρήσεως και ελεύθερης διανομής Λ/Σ LINUX που αντικατέστησε στα τέλη του 2003 το αντίστοιχο Λ/Σ Red Hat Linux. Η αντικατάσταση αυτή έγινε διότι η εταιρεία Red Hat ξεκίνησε την ανάπτυξη του Λ/Σ Red Hat Enterprise Linux, το οποίο είναι μεν γενικής χρήσεως, παρόλαυτά όμως η απόκτηση και η συντήρησή του έχει σημαντικό κόστος. Για τη μετάβαση από το ελεύθερο Λ/Σ Red Hat Linux στο ελεύθερο Λ/Σ Fedora Core, η Red Hat δημιούργησε και έκτοτε στηρίζει και χορηγεί την κοινότητα ανάπτυξης λογισμικού Fedora Project. Σήμερα το Λ/Σ Red Hat Enterprise Linux μπορεί να θεωρηθεί ως κλώνος των εκδόσεων του Fedora Core. Με αυτόν τον τρόπο η Red Hat έχει κατορθώσει να διαθέτει το Λ/Σ Fedora Core ως πρώτη ελεύθερη δοκιμή των επόμενων εκδόσεων του Λ/Σ Red Hat Enterprise Linux. Το όνομα Fedora αποτελεί σήμα κατατεθέν (στα αγγλικά trademark logo) της εταιρίας Red Hat.

3.4 Το Λ/Σ LINUX και ο βασικός Η/Υ της Διπλωματικής Εργασίας

Δεδομένου ότι δεν αναμένεται να διατηρηθεί για πολύ σε παραγωγική λειτουργία το σύστημα Η/Υ HP-9000/320, χρειάζεται η μετατροπή του υπάρχοντος ειδικού λογισμικού σειριακών επικοινωνιών (Χαρβάτης, 1990) ώστε ένα υλικό με το Λ/Σ

LINUX να μπορεί να συνδεθεί με όλους τους καταμετρητές του ΕΠΤ-ΕΜΠ μέσω σειριακής θύρας. Αυτή η μετατροπή θα επιτρέψει τη δόκιμη αντικατάσταση του συστήματος Η/Υ HP-9000/320 με υλικό που βασίζεται στην τεχνολογία των PC συμβατών υπολογιστών και έτσι οι μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI θα καταχωρούνται πλέον σε σύστημα Η/Υ για το οποίο θα υπάρχει εγγυημένη συντήρηση υλικού και λογισμικού.

Για τη μετατροπή του υπόψιν λογισμικού στα πλαίσια της παρούσης Διπλωματικής Εργασίας αποφασίσθηκε, μετά από έρευνα στον δικτυακό τόπο http://fedoraproject.org/ να χρησιμοποιηθεί η έκδοση Fedora Core 8 (Werewolf) του Λ/Σ LINUX. Η επιλογή αυτή έγινε για τους εξής λόγους:

(α) την εποχή που ανατέθηκε η Διπλωματική Εργασία στον γράφοντα (εαρινό εξάμηνο 2008), αυτή ήταν η πιο πρόσφατη έκδοση του Λ/Σ Fedora Core

(β) υπήρχε διαθέσιμο λογισμικό εγκατάστασης (drivers) για υλικό (κάρτες PCI) σειριακής επικοινωνίας και αναλογοψηφιακής μετατροπής της εταιρείας ADVANTECH (βλ. και http://www.advantech.com), το οποίο υπάρχει σε αφθονία στο ΕΠΤ-ΕΜΠ

(γ) υπήρχε "προς τα πίσω συμβατότητα" (στα αγγλικά backward compatibility) με την έκδοση Fedora Core 4, η οποία είναι ήδη εγκατεστημένη σε άλλα συστήματα στο ΕΠΤ-ΕΜΠ.

Ο Η/Υ στον οποίο επιλέχθηκε να εγκατασταθεί το Λ/Σ Fedora Core 8, ήταν ένα PC συμβατό, τα γενικά χαρακτηριστικά του οποίου έχουν ως εξής:

- μητρική ASUS P2B, chipset 08/01/2002-i440BX, ACPI BIOS revision 1014 beta 003 (με 2 σειριακές θύρες RS-232 9pin, 2 USB θύρες 1.0, 1 παράλληλη θύρα 25 pin)
- επεξεργαστής Intel Pentium II στα 350 MHz
- μνήμη SDRAM 224MB PC133 (1 x 32MB + 1 x 64MB + 1 x 128MB)
- κάρτα οθόνης VGA AGP x2, S3 Trio3D/2X, 4MB
- DVD-ROM (no name)
- σκληρό δίσκο χωρητικότητας 40GB (Western Digital WD400JB-00JJC0)

- κάρτα δικτύου 3COM τύπου 3C905B-TX

Το Λ/Σ Fedora Core 8, αποκτήθηκε από το διαδίκτυο σε μορφή ".iso", και μεταγράφηκε σε DVD, το οποίο επισυνάπτεται στη παρούσα Διπλωματική Εργασία[1]. Για να αρχίσει η εγκατάσταση, επιλέχθηκε από το BIOS του Η/Υ ως "1st Boot Device" το DVD-ROM (ή ισοδύναμα το CD-ROM). Στη συνέχεια τοποθετήθηκε στο DVD-ROM το DVD "Fedora-8-i386-DVD.iso" και επανεκκίνησε ο Η/Υ. Τα βήματα της εγκατάστασης ήταν χονδρικά τα εξής:

- έγινε boot από το DVD-ROM
- ο οδηγός εγκατάστασης πρότεινε να γίνει έλεγχος για να διαπιστωθεί ότι δεν λείπουν αρχεία ή ότι δεν υπάρχουν εσφαλμένα αρχεία στο DVD
- ο έλεγχος έγινε και αποδείχθηκε επιτυχής
- επιλέχθηκε η Αγγλική ως γλώσσα εγκατάστασης με τον ελληνικό τύπο πληκτρολογίου.
- επιλέχθηκε ο τρόπος με τον οποίο θα είναι οργανωμένος ο διαθέσιμος δίσκος, δηλαδή επιλέχθηκε πόση χωρητικότητα του δίσκου θα χρησιμοποιείται για τα αρχεία του Λ/Σ και των χρηστών αλλά και πόση χωρητικότητα θα χρησιμοποιείται ως "πρόχειρο" (στα αγγλικά swap). Προτάθηκε από τον οδηγό εγκατάστασης να χρησιμοποιείται ως πρόχειρο 512MB από τα 40GB, αλλά τελικά επιλέχθηκαν 1024MB για αυτό το σκοπό, προκειμένου να διευκολύνεται καλύτερα η λειτουργία του συστήματος. Προφανώς η απομένουσα χωρητικότητα διατέθηκε για τα αρχεία του Λ/Σ και των χρηστών.
- επιλέχθηκε το σύνθημα (στα αγγλικά password) του διαχειριστή του συστήματος (χρήστης root).
- έγινε αυτόματα η αντιγραφή των αναγκαίων αρχείων από το DVD-ROM στο σκληρό δίσκο

[1] Κατά τη συγγραφή του παρόντος κειμένου (2008) το Λ/Σ Fedora Core 8 υπήρχε διαθέσιμο και από τον ελληνικό σύνδεσμο ftp://ftp.otenet.gr/pub/linux/fedora/linux/releases/8/Fedora/i386/iso/ ως Fedora-8-i386-DVD.iso. Σημειώνεται ότι γενικά οι σύνδεσμοι του διαδικτύου για λογισμικό έχουν συνήθως μικρή διάρκεια ζωής.

- πραγματοποιήθηκαν κάποιες ρυθμίσεις που αφορούσαν στον τόπο λειτουργίας του συστήματος, στην ημερομηνία, στην ώρα και στο τείχος προστασίας

- ορίσθηκαν οι χρήστες που θα εργάζονται σε αυτό το σύστημα.

Μετά το τέλος της εγκατάστασης του Λ/Σ, κρίθηκε σκόπιμο ο Η/Υ να διασυνδεθεί με τον διαδίκτυο και για αυτό το λόγο του αποδόθηκε προσωρινά η ηλεκτρονική διεύθυνση 147.102.43.44 (του υποδικτύου "43" που ανήκει στο ΕΠΤ-ΕΜΠ). Επίσης τροποποιήθηκε το αρχείο /etc/rc.local ώστε να συμπεριλάβει την εντολή

/sbin/modprobe pcspkr

με τη βοήθεια της οποίας, κατά την εκκίνηση του συστήματος ενεργοποιείται το "καμπανάκι" (bell) του τερματικού.

Το Λ/Σ Fedora Core 8 διαθέτει εξ'ορισμού ως λειτουργικό σύστημα – κλώνος του αρχικά σχεδιασθέντος LINUX, την ομάδα μεταφραστών (στα αγγλικά compilers) γλωσσών προγραμματισμού gcc (gNU compiler collection), έκδοση 4.1.2, για γλώσσες προγραμματισμού C, C++, FORTRAN95, Java και άλλες. Σημειώνεται ότι ο μεταφραστής της FORTRAN95 μπορεί να κληθεί και με την εντολή gfortran. Από την εμπειρία που αποκτήθηκε από την εγκατάσταση του Λ/Σ Fedora Core 4, διαπιστώθηκε στο ΕΠΤ-ΕΜΠ ότι η γλώσσα FORTRAN95, όπως υπάρχει στην ομάδα μεταφραστών gcc στα Λ/Σ FEDORA, δεν είναι συμβατή με τη γλώσσα FORTRAN/9000, όπως αυτή υπήρχε στα συστήματα Η/Υ HP-9000/320-370 κάτω από το Λ/Σ HP-UX (βλ. και Hewlett Packard, 1987). Μετά από έρευνα βρέθηκε ότι καλύτερη συμβατότητα υπάρχει αν χρησιμοποιείται μεταφραστής για γλώσσα FORTRAN77 (Καράγγελος, 2007), ο οποίος είναι διαθέσιμος σε τρία απαραίτητα τμήματα (αρχεία rpm), ως εξής:

(α) compat-libf2c-34-3.4.6-8.i386.rpm

(β) compat-gcc-34-3.4.6-8.i386.rpm

(γ) compat-gcc-34-g77-3.4.6-8.i386.rpm.

από το δικτυακό τόπο http://rpm.pbone.net, στους ακόλουθους συνδέσμους:

- http://rpm.pbone.net/index.php3/stat/4/idpl/5326281/com/compat-libf2c-34-3.4.6-8.i386.rpm.html

- http://rpm.pbone.net/index.php3/stat/4/idpl/5312439/com/compat-gcc-34-3.4.6-8.i386.rpm.html
- http://rpm.pbone.net/index.php3/stat/4/idpl/5312441/com/compat-gcc-34-g77-3.4.6-8.i386.rpm.html

Τα τρία απαραίτητα αρχεία για τον υπόψιν μεταφραστή για γλώσσα FORTRAN77 περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία.

Προκειμένου να ενσωματωθεί και αυτός ο μεταφραστής στο Λ/Σ Fedora Core 8, τα σχετικά τρία πιο πάνω αρχεία αποθηκεύτηκαν στο Desktop του root και εγκαταστάθηκαν ένα προς ένα διαδοχικά από παραθυρικό περιβάλλον ακολουθώντας τη διαδικασία "double-click" από τον διαχειριστή του συστήματος. Σημειώνεται ότι κατά τη διάρκεια της εγκατάστασης μπορεί να ζητηθεί από τον οδηγό εγκατάστασης το DVD του Λ/Σ Fedora Core 8.

Επίσης, προκειμένου να είναι διαθέσιμοι και οι δύο μεταφραστές FORTRAN στον χρήστη με εύκολα διακριτό τρόπο προστέθηκαν στο φάκελο /usr/bin δύο συμβολικές συνδέσεις με τις ακόλουθες εντολές από τον διαχειριστή του συστήματος:

```
# ln –s /usr/bin/g77 /usr/bin/f77

# ln –s /usr/bin/gfortran /usr/bin/f95
```

Με αυτόν τον τρόπο ο μεταφραστής g77 της FORTRAN77 μπορεί να κληθεί και ως f77, ενώ ο μεταφραστής gfortran μπορεί να κληθεί ως f95.

Όσο αφορά στα ηλεκτρονικά εγχειρίδια (man pages) των δύο αυτών μεταφραστών, προκειμένου να υπάρχει εύκολη πρόσβαση σε αυτά από τον χρήστη και σύμφωνη με τις πιοπάνω συμβολικές συνδέσεις προστέθηκαν στο φάκελο /usr/share/man/man1 δύο συμβολικές συνδέσεις με τις ακόλουθες εντολές από τον διαχειριστή του συστήματος:

```
# ln –s g77.1.gz f77.1.gz, και

# ln –s gfortran.1.gz f95.1.gz
```

Για περαιτέρω τεκμηρίωση του μεταφραστή g77, μπορεί κανείς να μελετήσει:

```
#info g77
```

ή ισοδύναμα από το διαδίκτυο το:

http://gcc.gnu.org/onlinedocs/gcc-3.4.6/g77/

Τέλος, διαδοχικές επανεκκινήσεις του συστήματος Η/Υ με το Λ/Σ Fedora 8, όπως περιγράφηκε στα προηγούμενα, έδειξαν ότι το σύστημα εκκινούσε με καθυστέρηση. Παρακολουθώντας τη διαδικασία εκκίνησης διαπιστώθηκε ότι η αιτία της καθυστέρησης ήταν κυρίως η εκκίνηση της υπηρεσίας (στα αγγλικά service) sendmail και της υπηρεσίας sm-client. Κατά συνέπεια, αποφασίσθηκε να αδρανοποιηθούν οι υπηρεσίες αυτές με τις ακόλουθες εντολές από τον διαχειριστή του συστήματος:

```
# chkconfig sendmail off
# chkonfig sm-client off
```

3.5 Σειριακή θύρα

Από την εποχή της δημιουργίας των Η/Υ με την αρχιτεκτονική και τη δομή που είναι γνωστή και σήμερα, παρουσιάσθηκε η ανάγκη επικοινωνίας τους με άλλους Η/Υ ή με διάφορες συσκευές συλλογής δεδομένων. Για το σκοπό αυτό δημιουργήθηκαν οι λεγόμενες "θύρες επικοινωνίας". Με τη χρήση των θυρών αυτών η μετάδοση δεδομένων μεταξύ Η/Υ ή μεταξύ Η/Υ και περιφερειακών συσκευών (όπως είναι τα τερματικά, οι εκτυπωτές, οι μονάδες μαγνητικού δίσκου και λοιπές ψηφιακές μονάδες) μπορεί να γίνει με δύο τρόπους:

- Παράλληλα
- Σειριακά

Κατά την παράλληλη σύνδεση, όλα τα δυαδικά ψηφία (bits) που συγκροτούν μία "λέξη" της μνήμης του Η/Υ μεταδίδονται ταυτόχρονα και διαμέσου ξεχωριστών ενσύρματων γραμμών μίας για κάθε ένα bit. Στην απλούστερη περίπτωση, η "λέξη" είναι το byte που αποτελείται από 8 bits, συχνά όμως μπορεί μία λέξη να αποτελείται από 16 ή και 32 bits. Κυριότερο πλεονέκτημα της παράλληλης σύνδεσης, είναι η μεγάλη ταχύτητα μετάδοσης δεδομένων. Μειονεκτήματα είναι το μεγάλο πλήθος ενσύρματων γραμμών που απαιτούνται και το ότι είναι κατάλληλη μόνο για πολύ μικρές αποστάσεις. Κατά την σειριακή σύνδεση, για να επικοινωνήσει ο πομπός με τον δέκτη χρησιμοποιείται μία ενσύρματη γραμμή για κάθε μία κατεύθυνση μέσω της οποίας μεταδίδονται διαδοχικά ένα - ένα τα bits του πομπού προς τον δέκτη.

Πιοκάτω αναφέρονται ονομαστικά οι σημαντικότερες σήμερα χρησιμοποιούμενες θύρες επικοινωνίας, με ιστορική σειρά εμφάνισης:

"Σειριακή", "Παράλληλη", "GPIB / HPIB", "SCSI", "Ethernet", "USB" και "Firewire"

Ως πρώτη ιστορικά, και για αρκετό καιρό (έως περίπου το 1985) μόνη διαθέσιμη, η σειριακή θύρα (στα αγγλικά serial port), χρησιμοποιήθηκε πολύ για τη σύνδεση Η/Υ μεταξύ τους αλλά και με πειραματικές διατάξεις, όπως π.χ. γίνεται και στο ΕΠΤ-ΕΜΠ. Σήμερα η χρήση της δεν είναι τόσο συχνή όσο στο παρελθόν δεδομένου ότι για το σκοπό αυτό προσφέρονται σήμερα καλύτερα και με μεγαλύτερη αξιοπιστία κυρίως οι θύρες USB και Ethernet. Μάλιστα διαπιστώνεται μία ισχυρή τάση, κατάργησης αυτού του είδους θύρας ως ενσωματωμένες στις μητρικές κάρτες υπολογιστών τύπου PC συμβατών. Παρόλαυτά, υπάρχουν στην βιομηχανία, στα ερευνητικά εργαστήρια και στην καθημερινή πράξη πολλές συσκευές που διαθέτουν σειριακές θύρες επικοινωνίας. Επισημαίνεται επίσης ότι η κατασκευή μετρητικών συστημάτων που βασίζονται στη σειριακή επικοινωνία με Η/Υ δεν έχει σταματήσει. Οι ανάγκες επικοινωνίας όλων αυτών των συστημάτων με Η/Υ τύπου PC συμβατού, σε περίπτωση που δεν διατίθενται σειριακές θύρες στη μητρική κάρτα, καλύπτονται από μετατροπείς USB-to-Serial, ή από κατάλληλες κάρτες επέκτασης τεχνολογίας PCI. Η ταχύτητα μετάδοσης των πληροφοριών στη σειριακή σύνδεση μετριέται σε baud (για τις επικοινωνίες ψηφιακών υπολογιστών, η μονάδα baud είναι ταυτόσημη με τη μονάδα bits per second ή bps). Η μέθοδος αυτή είναι κατάλληλη για επικοινωνίες περιορισμένης απόστασης, η τιμή της οποίας εξαρτάται και από την ταχύτητα μεταφοράς δεδομένων σε bps. Για την τυπική ταχύτητα των 9600 bps, η μέγιστη απόσταση εκτιμάται σε 15 m περίπου. Για μεγαλύτερες αποστάσεις πρέπει να χρησιμοποιούνται κατάλληλοι ενισχυτές – διαμορφωτές.

Για κάθε μεταφερόμενο χαρακτήρα ASCII, η ακολουθία των bit που μεταδίδονται σε μία σειριακή επικοινωνία είναι :

- Ενα start bit
- Τα data bits του χαρακτήρα (συνήθως 7 ή 8) αρχίζοντας από το "λιγότερο σημαντικό" (στα αγγλικά Least Significant Bit ή LSB) και καταλήγοντας στο "πλέον σημαντικό" (στα αγγλικά Most Significant Bit ή MSB)
- Ενα parity bit (προαιρετικό). Η τιμή του parity bit είναι τέτοια ώστε τα data bits συν το parity να περιέχουν ζυγό (περίπτωση Even Parity) ή μονό (περίπτωση Odd Parity) αριθμό μονάδων. Το bit αυτό χρησιμεύει στον εντοπισμό λαθών που πιθανόν να σημειωθούν κατά τη μετάδοση του χαρακτήρα.

- Ένα ή δύο stop bits

Οι σειριακές θύρες απαντώνται σε δύο είδη συνδέσεων κατά την τυποποίηση EIA232 (Electronic Industries Association 232) ή RS232 (Recommended Standard 232), όπως είναι ευρύτερα γνωστή:

(α) με εννέα ακίδες (9 pin) σε βύσμα τύπου Amphenol type D (αλλιώς DB9)

(β) με εικοσιπέντε ακίδες (25 pin) σε βύσμα τύπου Amphenol type D (αλλιώς DB25)

Όπως είναι λογικό, κάθε pin αναφέρεται σε τυποποιημένο σήμα, έτσι ώστε να μπορούν οι θύρες από διαφορετικούς κατασκευαστές να συνεργάζονται. Από τα δύο είδη συνδέσεων, σήμερα στην πράξη διατηρείται μόνο η σύνδεση (α), δεδομένου ότι η σύνδεση (β) είναι κατάλληλη και για σύγχρονη επικοινωνία, της οποίας η χρήση έχει ατονήσει. Σημειώνεται όμως ότι στην αγορά διατίθενται και τα δύο είδη βυσμάτων, με τη διαφορά ότι συνήθως ακόμα και τα βύσματα με 25 pin, έχουν ενεργοποιημένα μόνο τα 9. Τα βύσματα αυτά αν βρίσκονται πάνω στον "πομπό" δεδομένων (π.χ. στον Η/Υ, ή πιο σωστά στο λεγόμενο Data Terminal Equipment ή DTE), είναι αρσενικά. Η τυποποίηση των εννέα απαραιτήτων σημάτων DTE και των αντιστοίχων pins σε αρσενικά βύσματα τύπου DB25 και DB9 διακρίνεται στον Πίνακα 3.1, ο οποίος πρέπει να εξετάζεται σε συνδυασμό με τα σχήματα του Διαγράμματος 3.2, που παρουσιάζουν αρσενικά βύσματα τύπου DB25 και DB9 τύπου DTE.

Αναλυτικότερα:

- Από το pin 2 του βύσματος DB25 ή το pin 3 του βύσματος DB9 αποστέλλονται δεδομένα.
- Στο pin 3 του βύσματος DB25 ή στο pin 2 του βύσματος DB9 λαμβάνονται δεδομένα.
- Αν το pin 20 του βύσματος DB25 ή το pin 4 του βύσματος DB9 έχει τάση, τότε η σειριακή θύρα είναι έτοιμη για μετάδοση δεδομένων.
- Αν το pin 6 του βύσματος DB25 ή το pin 6 του βύσματος DB9 έχει τάση, τότε η σειριακή θύρα πρέπει να ετοιμασθεί για παραλαβή δεδομένων.
- Αν το pin 4 του βύσματος DB25 ή το pin 7 του βύσματος DB9 έχει τάση, τότε η σειριακή θύρα ζητάει να αποστείλει δεδομένα.

- Αν το pin 5 του βύσματος DB25 ή το pin 8 του βύσματος DB9 έχει τάση, τότε η σειριακή θύρα επιτρέπει την παραλαβή δεδομένων διότι έχει αναγνωρίσει την αίτηση αποστολής δεδομένων από άλλη θύρα.

Κατά την σειριακή σύνδεση, για να επικοινωνήσουν ο πομπός (δηλαδή η συσκευή DTE) με τον δέκτη των δεδομένων (π.χ. έναν εκτυπωτή ή ένα modem, ή πιο σωστά με το λεγόμενο Data Circuit terminating Equipment ή DCE) χρησιμοποιείται μία ενσύρματη γραμμή για κάθε μία κατεύθυνση μέσω της οποίας μεταδίδονται διαδοχικά ένα - ένα τα bits του εξοπλισμού DTE προς τον εξοπλισμό DCE. Τα αντίστοιχα βύσματα τύπου DB25 ή DB9 που βρίσκονται πάνω στον "δέκτη" δεδομένων είναι θηλυκά. Η τυποποίηση των εννέα απαραιτήτων σημάτων DCE και των αντιστοίχων pins σε θηλυκά βύσματα τύπου DB25 και DB9 διακρίνεται στον Πίνακα 3.2, ο οποίος πρέπει να εξετάζεται σε συνδυασμό με τον Πίνακα 3.1 και με τα σχήματα των Διαγραμμάτων 3.3 έως και 3.5, που παρουσιάζουν διάφορες σειριακές συνδέσεις μεταξύ συσκευών DTE και DCE (βλ. και "Παραδείγματα" πιοκάτω στο κείμενο).

Ο τρόπος επικοινωνίας μέσω σειριακής σύνδεσης, διακρίνεται σε σύγχρονη και ασύγχρονη επικοινωνία. Κατά τη σύγχρονη επικοινωνία το ποιος εκπέμπει (δηλαδή το ποιος είναι εκείνη τη στιγμή DTE) και το ποιος δέχεται (δηλαδή το ποιος είναι εκείνη τη στιγμή DCE) από τις δύο συσκευές που επικοινωνούν κάθε φορά ρυθμίζεται από κατάλληλο χρονισμό. Η σύγχρονη επικοινωνία δεν είναι εξυπηρετική στην πράξη διότι ο χρήστης του συστήματος δεν έχει την δυνατότητα να παρέμβει για να την αλλάξει. Κατά την ασύγχρονη επικοινωνία ο εξοπλισμός DTE δεν πρέπει να στέλνει πληροφορίες αν ο εξοπλισμός DCE, δεν είναι έτοιμος να τις παραλάβει και αντίστροφα. Ο μηχανισμός με τον οποίο ο εξοπλισμός DTE πληροφορείται την κατάσταση του εξοπλισμού DCE (αν είναι έτοιμος ή όχι) ονομάζεται handshaking και υπάρχουν δύο τρόποι να υλοποιηθεί :

- με τη χρησιμοποίηση ενός πρωτοκόλλου επικοινωνίας (software handshaking). Το πρωτόκολλο επικοινωνίας είναι ένας προσυμφωνημένος τρόπος σύμφωνα με τον οποίο θα πρέπει να διεξάγεται η επικοινωνία και ο οποίος ορίζει πότε έχει δικαίωμα ο DTE ή ο DCE να στείλει πληροφορίες. Ένα ευρέως χρησιμοποιούμενο τέτοιο πρωτόκολλο είναι το πρωτόκολλο XON-XOFF σύμφωνα με το οποίο ο DTE διακόπτει την αποστολή χαρακτήρων αν του έρθει ο

ASCII χαρακτήρας DC3 (XOFF) και την ξαναρχίζει όταν του έρθει ο ASCII χαρακτήρας DC1 (XON).

- με τη χρησιμοποίηση κατάλληλης καλωδίωσης (hardware handshaking).

Για ασύγχρονη σειριακή σύνδεση διπλής κατεύθυνσης, με software handshaking απαιτούνται δύο ενσύρματες γραμμές (μία για κάθε κατεύθυνση) και μία γραμμή γείωσης.

Για ασύγχρονη σειριακή σύνδεση διπλής κατεύθυνσης, με hardware handshaking απαιτούνται δύο ενσύρματες γραμμές (μία για κάθε κατεύθυνση), μία γραμμή γείωσης και ορισμένες επιπλέον γραμμές ελέγχου.

Στην παρούσα Διπλωματική Εργασία, οι συνδέσεις που θα υλοποιηθούν θα είναι DTE – DCE, ασύγχρονης σειριακής σύνδεσης διπλής κατεύθυνσης, με software handshaking, για τις οποίες απαιτούνται προφανώς απλά καλώδια με τρία σύρματα.

Παραδείγματα:

Στο σχήμα του Διαγράμματος 3.3 φαίνεται ένα παράδειγμα συνήθους ασύγχρονης σειριακής σύνδεσης διπλής κατεύθυνσης, με hardware handshaking με βύσματα DB9. Απαιτούνται δύο ενσύρματες γραμμές (μία για κάθε κατεύθυνση), μία γραμμή γείωσης και ορισμένες επιπλέον γραμμές ελέγχου. Σύμφωνα με το παράδειγμα:

- Όταν η θύρα 1 στέλνει μέσω του pin 3, η θύρα 2 λαμβάνει μέσω του pin 2 και αντίστροφα.
- Η θύρα 1 γνωρίζει στη θύρα 2 ότι είναι έτοιμη για μετάδοση δεδομένων στέλνοντας από το δικό της pin 4 τάση στο pin 6 της θύρας 2 και αντίστροφα.
- Η θύρα 1 γνωρίζει στη θύρα 2 ότι ζητάει αποστολή δεδομένων στέλνοντας από το δικό της pin 7 τάση στο pin 8 της θύρας 2 και αντίστροφα.

Στο σχήμα του Διαγράμματος 3.4 φαίνεται ένα παράδειγμα συνήθους ασύγχρονης σειριακής σύνδεσης διπλής κατεύθυνσης, με software handshaking με βύσματα DB9. Απαιτούνται δύο ενσύρματες γραμμές (μία για κάθε κατεύθυνση) και μία γραμμή γείωσης. Σύμφωνα με το παράδειγμα:

- Όταν η θύρα 1 στέλνει μέσω του pin 3, η θύρα 2 λαμβάνει μέσω του pin 2 και αντίστροφα.
- Η θύρα 1 γνωρίζει πότε πρέπει να στείλει αν έχει λάβει τον χαρακτήρα XON

- Η θύρα 1 γνωρίζει πότε πρέπει να σταματήσει να στέλνει αν έχει λάβει τον χαρακτήρα XOFF.
- Η θύρα 2 γνωρίζει πότε πρέπει να στείλει αν έχει λάβει τον χαρακτήρα XON
- Η θύρα 2 γνωρίζει πότε πρέπει να σταματήσει να στέλνει αν έχει λάβει τον χαρακτήρα XOFF.

3.6 Επιτήρηση ASCII χαρακτήρων σειριακής επικοινωνίας

Για τις ανάγκες της σειριακής επικοινωνίας με Η/Υ, των μονάδων επικοινωνίας CANBERRA, πρέπει γενικά ο Η/Υ να αποστέλλει προς αυτές ASCII χαρακτήρες εντολών και να λαμβάνει από αυτές ASCII χαρακτήρες μετρήσεων. Αυτή η συναλλαγή θα είναι επιτυχής όταν από τον κώδικα του Η/Υ αποστέλλονται οι σωστοί χαρακτήρες προς τη μονάδα επικοινωνίας και λαμβάνονται από αυτή οι αναμενόμενοι χαρακτήρες που αντιπροσωπεύουν τις μετρήσεις. Για την επιβεβαίωση της ορθής επικοινωνίας υπάρχουν δύο τρόποι:

(α) ο έμμεσος, η προσεκτική δηλαδή παρακολούθηση των συνεπειών από τις αποστελλόμενες εντολές και η επαλήθευση των καταγραφόμενων αποτελεσμάτων μέσω διεξαγωγής πραγματικών πειραμάτων, και

(β) ο άμεσος, η "υποκλοπή" δηλαδή των αποστελλόμενων και λαμβανόμενων χαρακτήρων από το ίδιο το καλώδιο που εξυπηρετεί τη σειριακή επικοινωνία.

Ο τρόπος (β) είναι πολύ προτιμότερος διότι δίνει άμεση εποπτεία του τρόπου λειτουργίας του λογισμικού του Η/Υ και του τρόπου ανταπόκρισης της μονάδας επικοινωνίας σε αυτό.

Για την υλοποίηση του τρόπου (β) απαιτούνται δύο τεχνικές προϋποθέσεις:

(1) κατάλληλο καλώδιο "υποκλοπής"

(2) κατάλληλο λογισμικό και Η/Υ για την "υποκλοπή"

Αν το καλώδιο σειριακής επικοινωνίας είναι αυτό του σχήματος στο Διάγραμμα 3.5, τότε για την κατασκευή του καλωδίου "υποκλοπής" (στα αγγλικά spy cable) ακολουθήθηκε το σχήμα του Διαγράμματος 3.6. Το spy cable αντιπροσωπεύει μία παρεμβολή στο καλώδιο επικοινωνίας μεταξύ των θέσεων "FROM" και "TO". Δεδομένα που αποστέλλονται από το pin "Η/Υ 3" λαμβάνονται από το pin "ΜΟΝΑΔΑ ΕΠΙΚΟΙΝΩΝΙΑΣ 3" και "υποκλέπτονται" από το pin "SPY1 2".

Δεδομένα που αποστέλλονται από το pin "ΜΟΝΑΔΑ ΕΠΙΚΟΙΝΩΝΙΑΣ 2" λαμβάνονται από το pin "Η/Υ 2" και "υποκλέπτονται" από το pin "SPY2 2". Τα pin "5" και "7" αντιπροσωπεύουν το δυναμικό "0". Τα pins "SPY1 2" και "SPY2 2" οδηγούνται μέσω σειριακού καλωδίου σε δύο σειριακές θύρες του Η/Υ "υποκλοπής" (τις COM1 και COM2, με τον τρόπο που διακρίνεται στο σχήμα του Διαγράμματος 3.7), στον οποίο "τρέχει" το κατάλληλο λογισμικό "υποκλοπής". Η συνδεσμολογία του spy cable υλοποιήθηκε μέσα σε μικρό ηλεκτροστατικό κουτί, οι αναγκαίες συγκολλήσεις έγιναν σε ηλεκτρονική πλακέτα "γενικών κατασκευών" και η όλη κατασκευή παρουσιάζεται συνολικά στις φωτογραφίες των Διαγραμμάτων 3.8 και 3.9.

Ο υπολογιστής "υποκλοπής" ή από εδώ και πέρα το "spy PC" είναι ένα παλαιό (του 1998) PC συμβατό, τα γενικά χαρακτηριστικά του οποίου έχουν ως εξής:

- μητρική Intel MAUI3, BIOS Intel 4M4UE0X3.86A.0006.P05 12/04/98 (με 1 σειριακή θύρα RS-232 9pin, 2 USB θύρες 1.0, 1 παράλληλη θύρα 25 pin)
- επεξεργαστής Intel Celeron 333 MHz
- μνήμη RAM 64MB PC (2 x 32MB)
- κάρτα οθόνης ATI 3D Rage Pro
- CD-ROM (no name)
- σκληρό δίσκο χωρητικότητας 4.32 GB
- κάρτα I/O ISA BUS ZyXEL SP111AT (1 σειριακή θύρα RS-232 9pin, 1 παράλληλη θύρα 25 pin)
- Λ/Σ Microsoft Windows 98 SE (MS W98 SE)

Στο spy PC ως σειριακή θύρα COM1 αναγνωρίζεται η σειριακή RS-232 που υπάρχει ενσωματωμένη στη μητρική κάρτα. Για τη λειτουργία του spy cable απαιτείται φυσικά και δεύτερη σειριακή θύρα COM2. Αυτή μπορεί να προστεθεί στο spy PC ως περιφερειακή μονάδα. Στο ΕΠΤ-ΕΜΠ υπήρχε διαθέσιμη τέτοια περιφερειακή μονάδα η κάρτα I/O ISA BUS ZyXEL SP111AT, φωτογραφία της οποίας διακρίνεται στο Διάγραμμα 3.10, και η οποία ρυθμίσθηκε και τοποθετήθηκε στο spy PC, ως εξής:

Ρύθμιση IRQ:

Προκειμένου μία περιφερειακή μονάδα Η/Υ να μπορεί να επικοινωνήσει με την Κεντρική Μονάδα Επεξεργασίας (CPU), θα πρέπει να αποστείλει μία "αίτηση παρενθετικής διακοπής" (στα αγγλικά Interrupt ReQuest ή IRQ), με την οποία να ζητάει επικοινωνία. Στο επίπεδο που αντιλαμβάνεται ο χρήστης οι αιτήσεις αυτές αντιπροσωπεύονται με αριθμούς, οι οποίοι για ένα PC συμβατό, τεχνολογίας AT με επεξεργαστή Intel 80x86, έχουν συνήθως όπως ο Πίνακας 3.3. Οι αριθμοί αυτοί διατίθενται στις περιφερειακές μονάδες είτε αυτόματα μετά την αναγνώρισή τους από το λειτουργικό σύστημα, είτε με παρέμβαση του χρήστη, προτού αυτός συνδέσει την περιφερειακή μονάδα επικοινωνίας που τον ενδιαφέρει. Δεδομένου ότι οι περιφερειακές μονάδες ISA BUS, δεν είναι από αυτές που αναγνωρίζονται αυτόματα από το Λ/Σ (δεν είναι δηλαδή plug and play), για να συνδεθεί και να λειτουργήσει η κάρτα I/O ISA BUS ZyXEL SP111AT στο spy PC, απαιτήθηκε η εύρεση ελεύθερου IRQ και η ρύθμισή της σε αυτό. Προκειμένου να διαπιστωθεί ποιο IRQ είναι διαθέσιμο, αναζητήθηκαν τα ήδη δεσμευμένα με την ακόλουθη διαδικασία:

Από την επιφάνεια εργασίας με τη χρήση του "ποντικιού":

> My Computer > Δεξί Πλήκτρο > Properties > Device Manager > View Devices by Type, και στη συνέχεια αφού έγινε "expand tree" ο παρουσιαζόμενος πίνακας περιφερειακών σε όλες τις συνιστώσες του

Για κάθε περιφερειακό (Device) με τη χρήση του "ποντικιού":

> Επιλογή ("φωτισμός") > Δεξί πλήκτρο > Properties > (όπου διατίθεται) Resources

Από την καταγραφή διαπιστώθηκε ότι δεσμευμένα είναι τα IRQ 01 (πληκτρολόγιο), 04 (σειριακή θύρα COM1), 06 (Μονάδα Δισκέτας), 07 (παράλληλη πόρτα), 09 (κάρτα οθόνης, θύρες USB), 10 (κάρτα δικτύου), 12 (ποντίκι), 14 (σκληρός δίσκος) και 15 (σκληρός δίσκος). Έτσι, πριν την τοποθέτησή της, στην κάρτα I/O ISA BUS ZyXEL SP111AT, ρυθμίσθηκε το IRQ της σειριακής θύρας COM2 σε 5, με τη χρήση του "βραχυκυκλωτήρα" (στα αγγλικά jumper) JP1[4+5] = (1) (1), σύμφωνα και με τις οδηγίες του σχετικού εγχειριδίου της κάρτας στο Παράρτημα, το οποίο εντοπίσθηκε στον πιοκάτω σύνδεσμο:

http://us.zyxel.com/support/knowledgebase1.php?indexFlagvalue=1021705590&level=3&upFlag=1021683267

Ρύθμιση Input/Output Range:

Προκειμένου μία περιφερειακή μονάδα Η/Υ να μπορεί να λειτουργήσει χρειάζεται να είναι γνωστό σε ποια περιοχή διευθύνσεων της μνήμης (στα αγγλικά Input/Output Range) "γράφονται" τα δεδομένα, τα οποία πρέπει να στείλει προς την CPU ή προς άλλες περιφερειακές μονάδες ή τα δεδομένα τα οποία πρέπει να λάβει από την CPU ή από άλλες περιφερειακές μονάδες. Καλό είναι οι περιοχές διευθύνσεων μνήμης που απασχολεί μία περιφερειακή μονάδα να μην είναι ίδιες με εκείνες που απασχολεί άλλη περιφερειακή μονάδα. Δεδομένου ότι η περιοχή διευθύνσεων μνήμης που απασχολεί η σειριακή θύρα COM1 είναι από 03F8 έως 03FF (όπως διαπιστώθηκε από τα Resources του Device COM1, προηγούμενα), στην κάρτα I/O ISA BUS ZyXEL SP111AT, ρυθμίσθηκε η σειριακή θύρα COM2, για την διαφορετική περιοχή διευθύνσεων μνήμης 02F8 έως 02FF,.με τη χρήση του jumper JP1[6+7] = (0) (1), σύμφωνα και με τις οδηγίες του σχετικού εγχειριδίου της κάρτας στο Παράρτημα.

Τοποθέτηση και λειτουργία:

Η κάρτα I/O ISA BUS ZyXEL SP111AT τοποθετήθηκε σε ελεύθερη θέση ISA πάνω στην μητρική κάρτα σύμφωνα με τις φωτογραφίες του Διαγράμματος 3.11. Στη συνέχεια εκκίνησε ο Η/Υ και ακολουθήθηκε η διαδικασία χειροκίνητης εγκατάστασης υλικού, όπως προβλέπεται για το Λ/Σ MS W98 SE:

> Start > Settings > Control Panel > Add new hardware > Next > Next > {Do you want Windows to search for your new hardware? (•) Yes } > Next > Next > {To manually install a specific device, click Next} > Next > {Select the type of hardware you want to install, επιλέγεται , ["φωτίζεται"], Ports (COM + LPT)} > Next > Επιλέγονται ["φωτίζονται"] Standard Port Types / Communications Port > Next > Αυτόματα αναγνωρίζεται η περιοχή διευθύνσεων μνήμης, όπως διαβάζεται από την ρύθμιση της κάρτας ZyXEL SP111AT > Next > Finish > Επανεκκίνηση Η/Υ.

Μετά την επανεκκίνηση η Communication Port που προστέθηκε, αναγνωρίζεται ως σειριακή θύρα COM2. Αν τυχόν η ρύθμιση του IRQ και του Input/Output Range, όπως διαβάζονται από την κάρτα ZyXEL SP111AT δεν έχουν αναγνωρισθεί σωστά, τότε με τη γνωστή διαδικασία:

Από την επιφάνεια εργασίας με τη χρήση του "ποντικιού":

> My Computer > Δεξί Πλήκτρο > Properties > Device Manager > View Devices by Type, και στη συνέχεια αφού έγινε "expand tree" ο παρουσιαζόμενος πίνακας περιφερειακών σε όλες τις συνιστώσες του

Για τη σειριακή πόρτα COM2 με τη χρήση του "ποντικιού":

> Επιλογή ("φωτισμός") > Δεξί πλήκτρο > Properties > Resources > Change Setting

μπορούν να γίνονται οι αναγκαίες αλλαγές ώστε οι ρυθμίσεις της κάρτας να αντιστοιχίσουν στις ρυθμίσεις του Η/Υ. Πιθανόν να απαιτείται και πάλι επανεκκίνηση.

Για τη λειτουργία του spy cable, εκτός από το αναγκαίο υλικό, και τις σχετικές ρυθμίσεις του, οι οποίες περιγράφθηκαν πιοπάνω, απαιτήθηκε και κατάλληλο λογισμικό, το οποίο φυσικά έπρεπε να εγκατασταθεί στο spy PC. Ως τέτοιο επιλέχθηκε το ελεύθερο λογισμικό "Serial Receptor 1.4", όπως αυτό διατίθεται στο σύνδεσμο http://www.tucows.com/preview/4245508. Είναι ένα πρόγραμμα που δημιουργήθηκε για την παρακολούθηση δεδομένων που στέλνονται σε και λαμβάνονται από σειριακές θύρες. Το απαραίτητο εκτελέσιμο αρχείο εγκατάστασης για το λογισμικό αυτό (SRsetupUS.exe, 575.68 kB) περιέχεται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία. Η εγκατάστασή του "Serial Receptor 1.4" στο spy PC ήταν απλή και έγινε με τη διαδικασία "double-click" πάνω στο εκτελέσιμο αρχείο εγκατάστασης ακολουθώντας τις οδηγίες που δίνονται από τον οδηγό εγκατάστασης. Μετά από αυτή τη διαδικασία το λογισμικό καταχωρήθηκε στο λειτουργικό σύστημα MSW98 και μπορεί να ενεργοποιείται ως

Start > Programs > Serial Receptor > Serial Receptor

ή, καλύτερα, από αντίστοιχη συντόμευση (shortcut) πάνω στην επιφάνεια εργασίας (Desktop).

- Για την παρακολούθηση της διαδρομής DB9 > DB25 όπως αυτή διακρίνεται στο σχήμα του Διαγράμματος 3.5, χρησιμοποιείται η διαδρομή του spy cable FROM > SPY1 > COM1, όπως διακρίνεται στα σχήματα των Διαγραμμάτων 3.6 και 3.7. Η παρακολούθηση της σειριακής θύρας COM1 μπορεί να ξεκινήσει με τη διαδικασία "double-click" πάνω στην Desktop συντόμευση του Serial Receptor

[βλ. και την αντίστοιχη "εκτύπωση οθόνης" (print screen) στην εικόνα του Διαγράμματος 3.12]. Στο "παράθυρο" που ανοίγει (βλ. και το print screen του Διαγράμματος 3.13) απαιτείται ρύθμιση των παραμέτρων της σειριακής θύρας COM1, αν αυτές δεν είναι σύμφωνες με τις παραμέτρους επικοινωνίας των συσκευών που βρίσκονται συνδεδεμένες στις σειριακές θύρες DB9 και DB25. Οι παράμετροι αυτές είναι:

(α) η παρακολουθούμενη θύρα (εξ'ορισμού η COM1)

(β) η ταχύτητα επικοινωνίας (εξ'ορισμού 9600 baud)

(γ) μέγεθος μεταφερόμενου byte (εξ'ορισμού 8 bit)

(δ) ρύθμιση parity (εξ'ορισμού none) , και

(ε) πλήθος stop bit (εξ'ορισμού 1)

και διακρίνονται πάντα στο κάτω δεξιά μέρος του υπόψιν παραθύρου.

Η ρύθμιση των παραμέτρων αυτών γίνεται με τη διαδικασία Setup > Serial Port, όπως διακρίνεται στο print screen του Διαγράμματος 3.14, και από τις επιλογές που προκύπτουν σε pull down menu, όπως διακρίνεται στο print screen του Διαγράμματος 3.15.

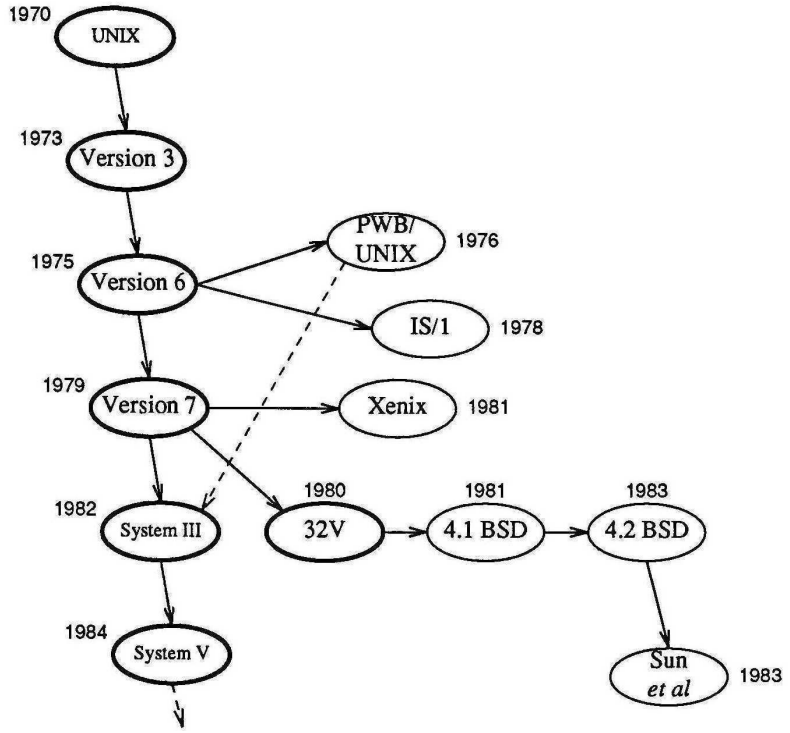
- Για την παρακολούθηση της διαδρομής DB25 > DB9 όπως αυτή διακρίνεται στο σχήμα του Διαγράμματος 3.5, χρησιμοποιείται η διαδρομή του spy cable TO > SPY2 > COM2, όπως διακρίνεται στα σχήματα των Διαγραμμάτων 3.6και 3.7. Η παρακολούθηση και η ρύθμιση της σειριακής θύρας COM2, αν ήδη παρακολουθείται η θύρα COM1 μπορεί να ξεκινήσει με τη διαδικασία File > New, όπως διακρίνεται στο print screen του Διαγράμματος 3.16, και από τις επιλογές που προκύπτουν σε pull down menu, όπως διακρίνεται στα print screen των Διαγραμμάτων 3.17 και 3.18. Ανοίγει τότε δεύτερο "παράθυρο" Untitled1, μέσα στο κεντρικό "παράθυρο" της εφαρμογής (βλ. και το print screen του Διαγράμματος 3.19).

3.7 Συμπεράσματα - Σχόλια

Στο Κεφάλαιο αυτό περιγράφθηκε με ικανοποιητική πληρότητα η συγκρότηση του απαραίτητου υλικού, ο τρόπος εγκατάστασης του απαιτούμενου Λ/Σ LINUX καθώς και του λοιπού απαραίτητου λογισμικού προκειμένου να εξυπηρετηθεί η μετατροπή

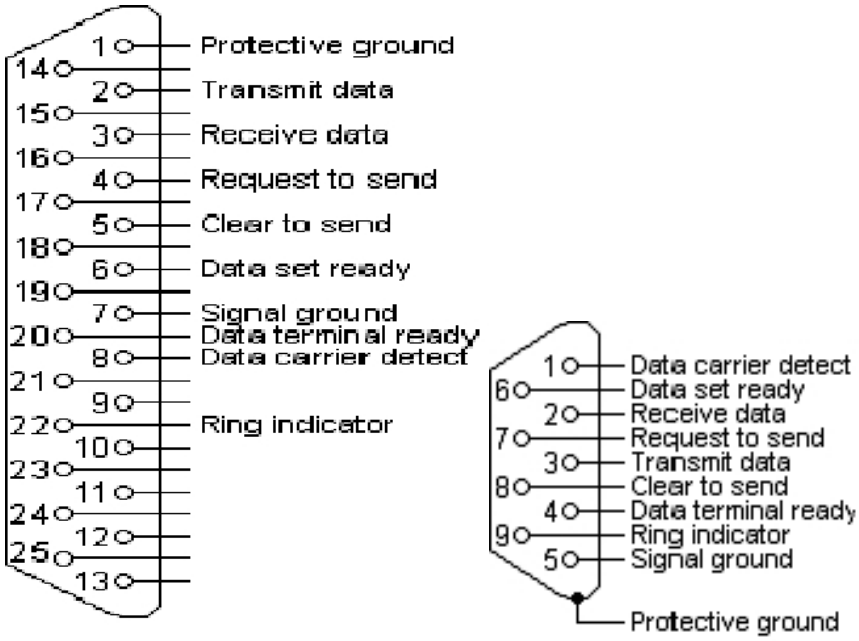
λογισμικού Λ/Σ UNIX SYSTEM V σε λογισμικό Λ/Σ LINUX για τη σύνδεση σειριακών συσκευών μέσω θύρας RS-232. Για το σκοπό αυτό καλύφθηκαν οι εξής ενότητες: (α) ιστορική αναδρομή σε προηγούμενες τεχνοτροπίες εφαρμογής τέτοιου λογισμικού σε άλλα Λ/Σ στο ΕΠΤ-ΕΜΠ, (β) ανασκόπηση της υπάρχουσας σήμερα (2008) κατάστασης σχετικά με το λογισμικό αυτό και την εφαρμογή του στο Λ/Σ UNIX SYSTEM V, (γ) μακροσκοπική περιγραφή του Λ/Σ UNIX SYSTEM V και της σταδιακής εξέλιξής του στο Λ/Σ LINUX, (δ) μέχρι σήμερα (2008) εφαρμογή του Λ/Σ LINUX στο ΕΠΤ-ΕΜΠ, (ε) περιγραφή του Η/Υ PC συμβατού και της έκδοσης του Λ/Σ LINUX που χρησιμοποιήθηκε, (στ) περιγραφή των τεχνικών και λογικών χαρακτηριστικών μιας σειριακής θύρας RS-232 και (ζ) περιγραφή του Η/Υ PC συμβατού και του ειδικού λογισμικού για MSW98, που χρησιμοποιήθηκε για την παρακολούθηση της μετάδοσης σειριακών δεδομένων από και προς κατάλληλες μονάδες επικοινωνίας. Στο επόμενο 4^{ο} Κεφάλαιο, παρουσιάζεται ο τρόπος με τον οποίο έγινε η μετατροπή λογισμικού Λ/Σ UNIX SYSTEM V σε λογισμικό Λ/Σ LINUX για τη σύνδεση σειριακών συσκευών μέσω θύρας RS-232.

ΔΙΑΓΡΑΜΜΑΤΑ ΤΟΥ 3ου ΚΕΦΑΛΑΙΟΥ



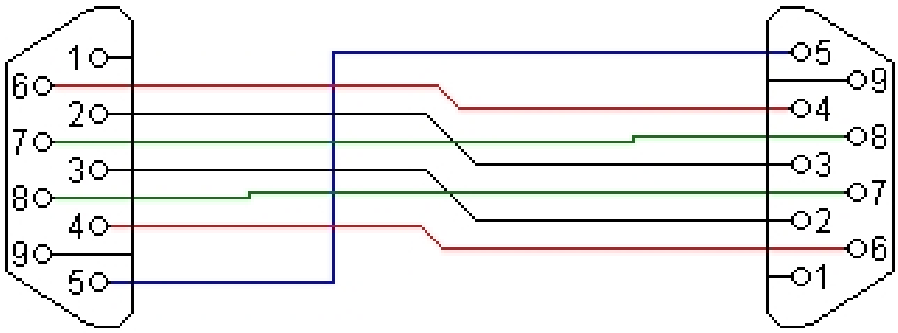
Διάγραμμα 3.1

Εξέλιξη του Λ/Σ UNIX από το 1970 μέχρι περίπου το 1984



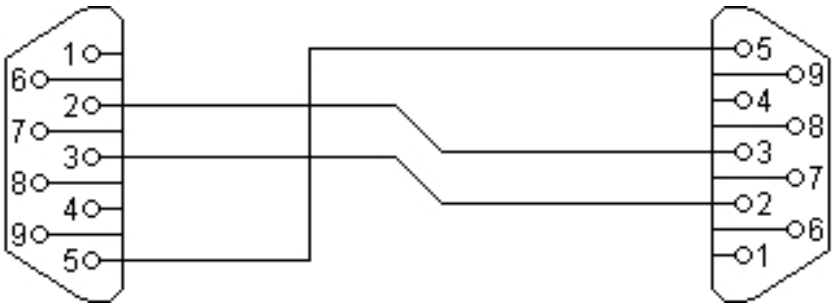
Διάγραμμα 3.2

Τυποποίηση εννέα απαραίτητων σημάτων και αντίστοιχα pins σε βύσματα τύπου DB25 και DB9, αρσενικά DTE



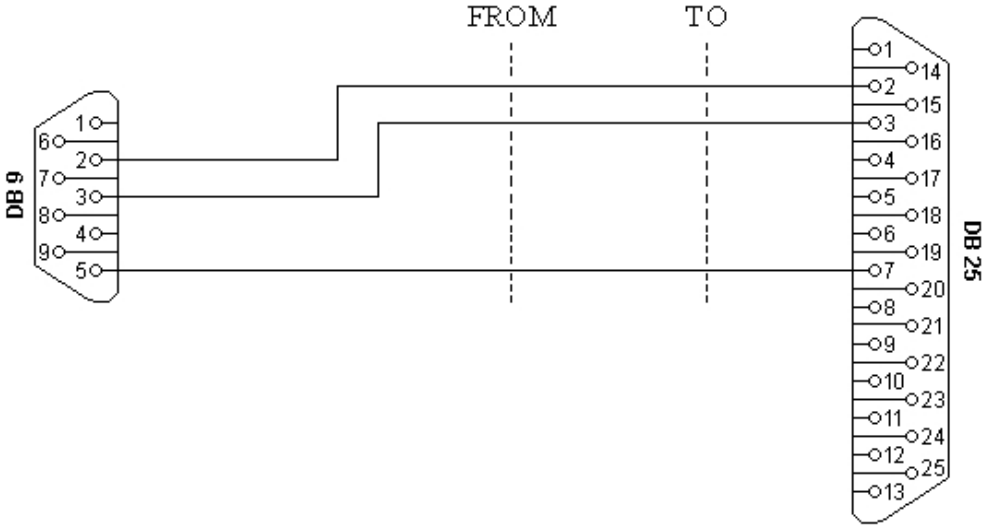
Διάγραμμα 3.3

Παράδειγμα συνήθους ασύγχρονης σειριακής σύνδεσης διπλής κατεύθυνσης, με δυνατότητα hardware handshaking με βύσματα DB9.
Αριστερά: το βύσμα DTE, δεξιά: το βύσμα DCE



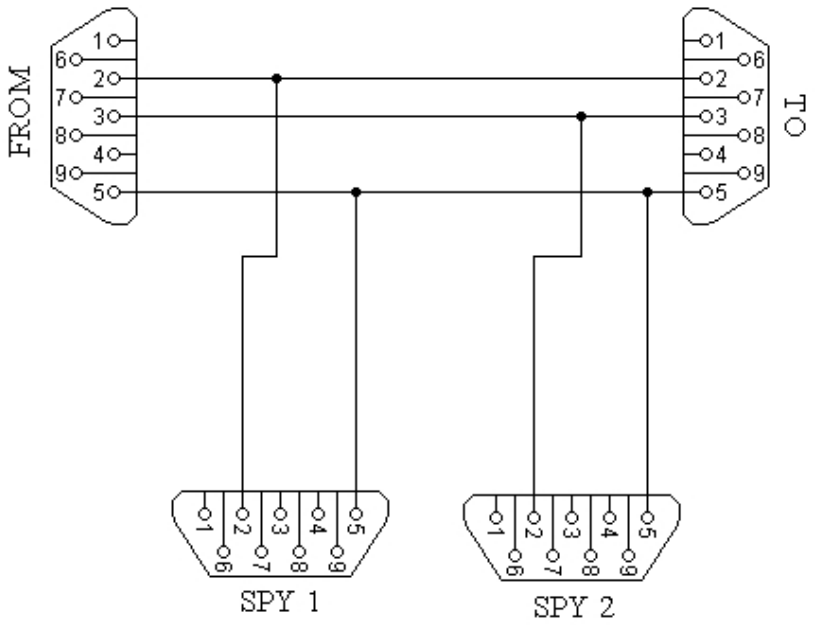
Διάγραμμα 3.4

Παράδειγμα συνήθους ασύγχρονης σειριακής σύνδεσης διπλής κατεύθυνσης, που μπορεί να λειτουργεί μόνο με software handshaking με βύσματα DB9.
Αριστερά: το βύσμα DTE, δεξιά: το βύσμα DCE



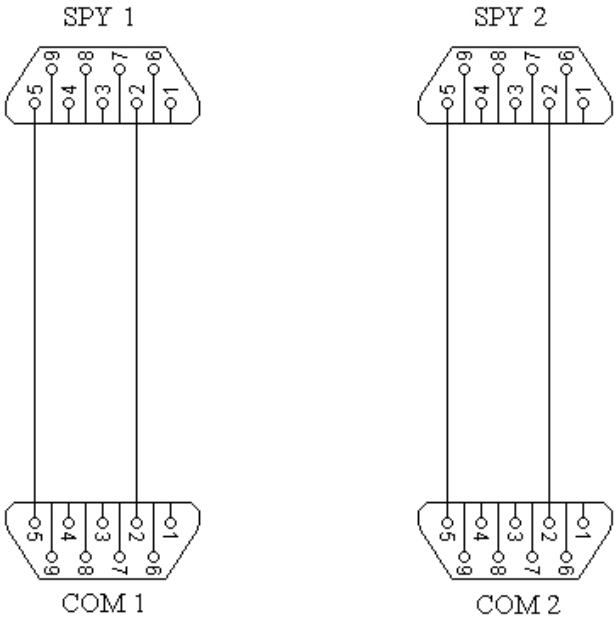
Διάγραμμα 3.5

Καλώδιο σειριακής επικοινωνίας DB9 – DB25 (software handshaking)
Αριστερά: το βύσμα DTE, δεξιά: το βύσμα DCE



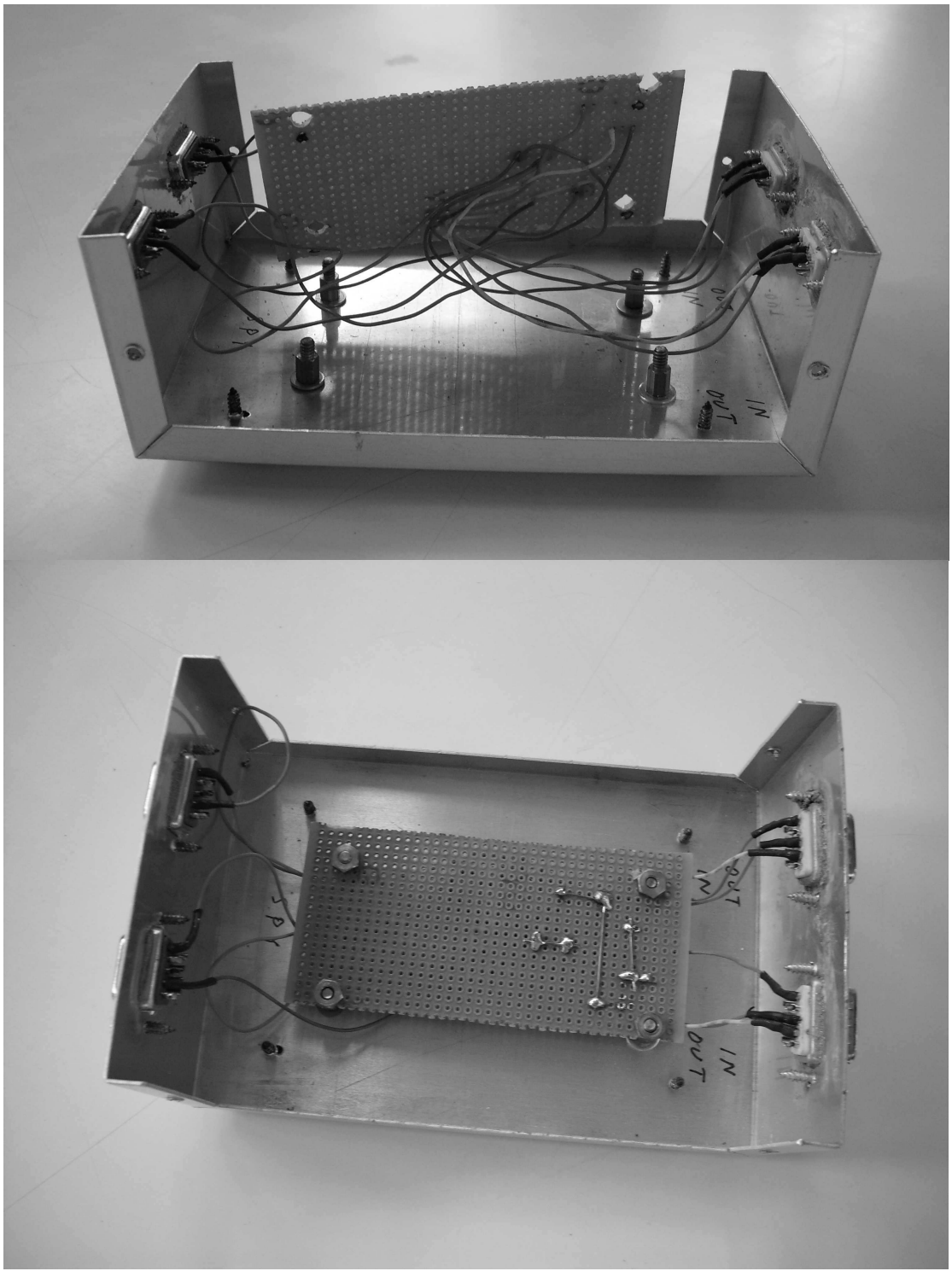
Διάγραμμα 3.6

Καλώδιο "υποκλοπής" (spy cable)



Διάγραμμα 3.7

Σύνδεση του καλωδίου "υποκλοπής" με τον Η/Υ "υποκλοπής" (spy PC)

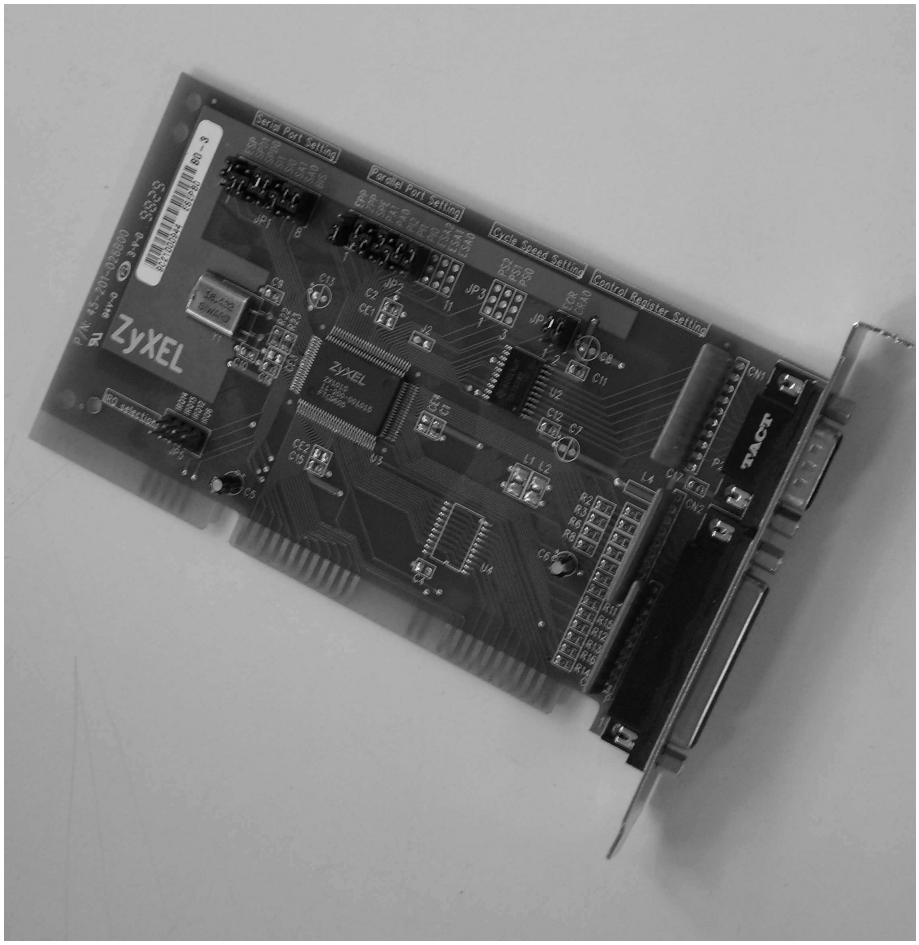


Διάγραμμα 3.8

Υλοποίηση spy cable, εσωτερικό ηλεκτροστατικού κουτιού, πλακέτα "γενικών κατασκευών"

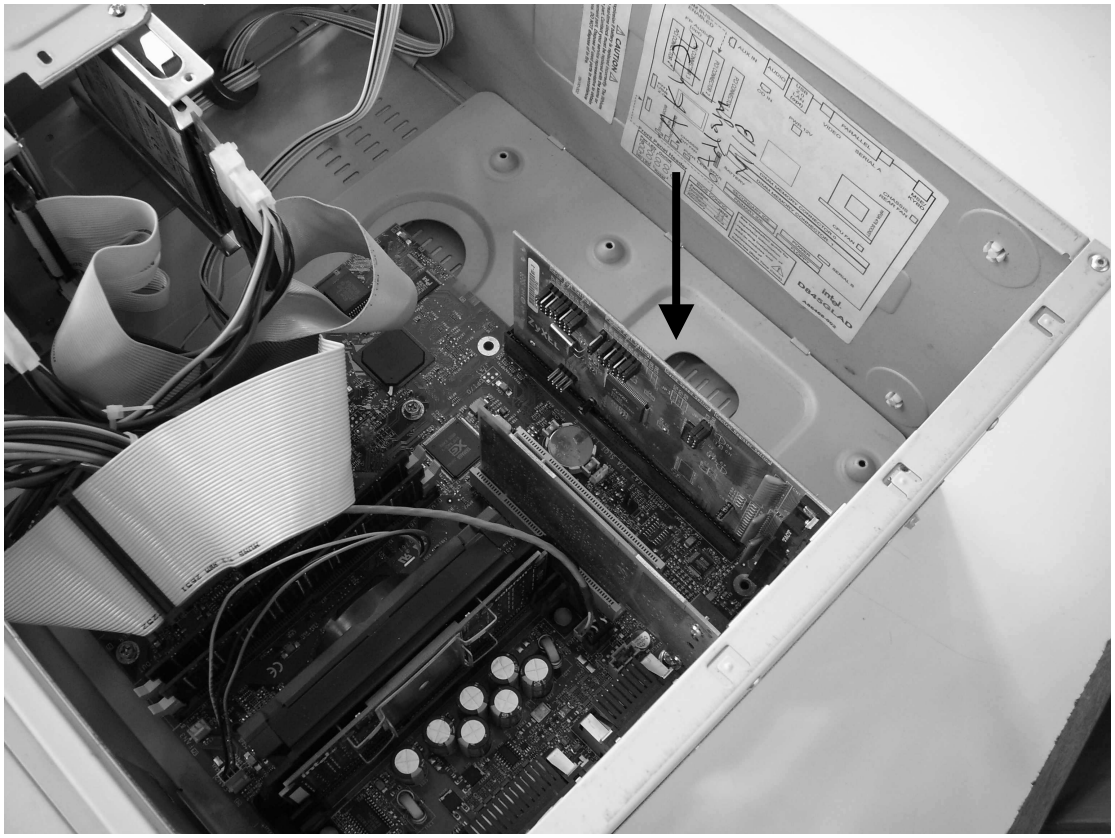
Διάγραμμα 3.9

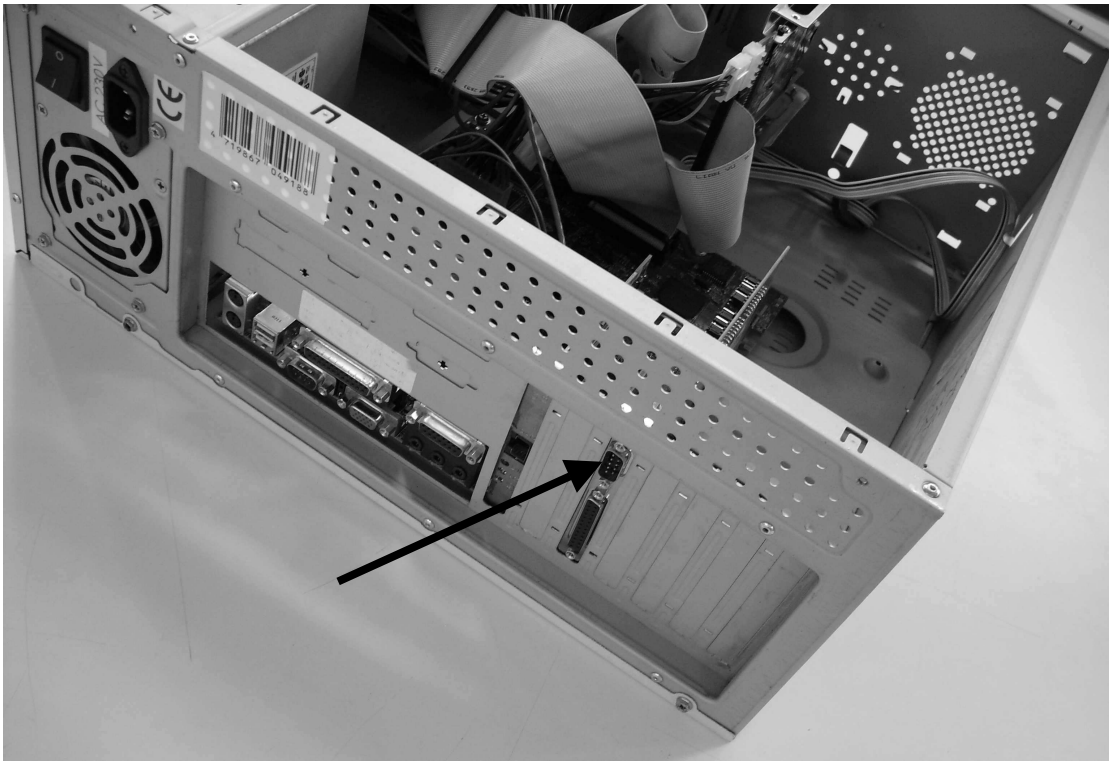
Υλοποίηση spy cable, εξωτερικό ηλεκτροστατικού κουτιού, "εμπρός" και "πίσω" όψη



Διάγραμμα 3.10

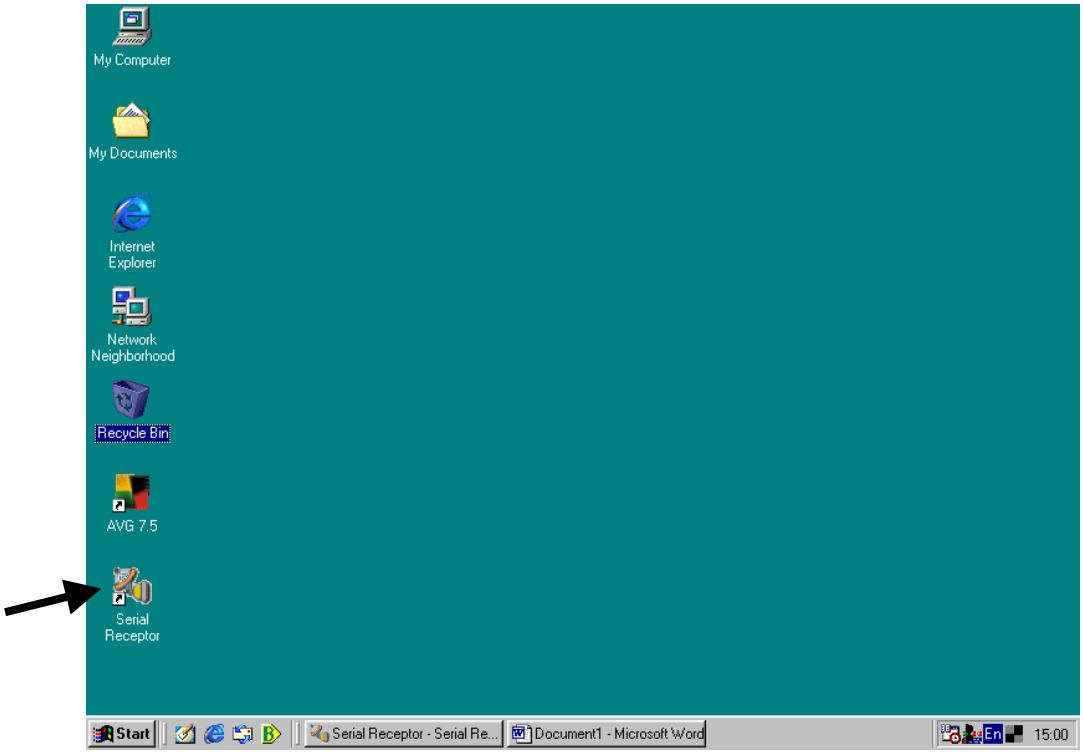
Περιφερειακή μονάδα κάρτα I/O ISA BUS ZyXEL SP111AT





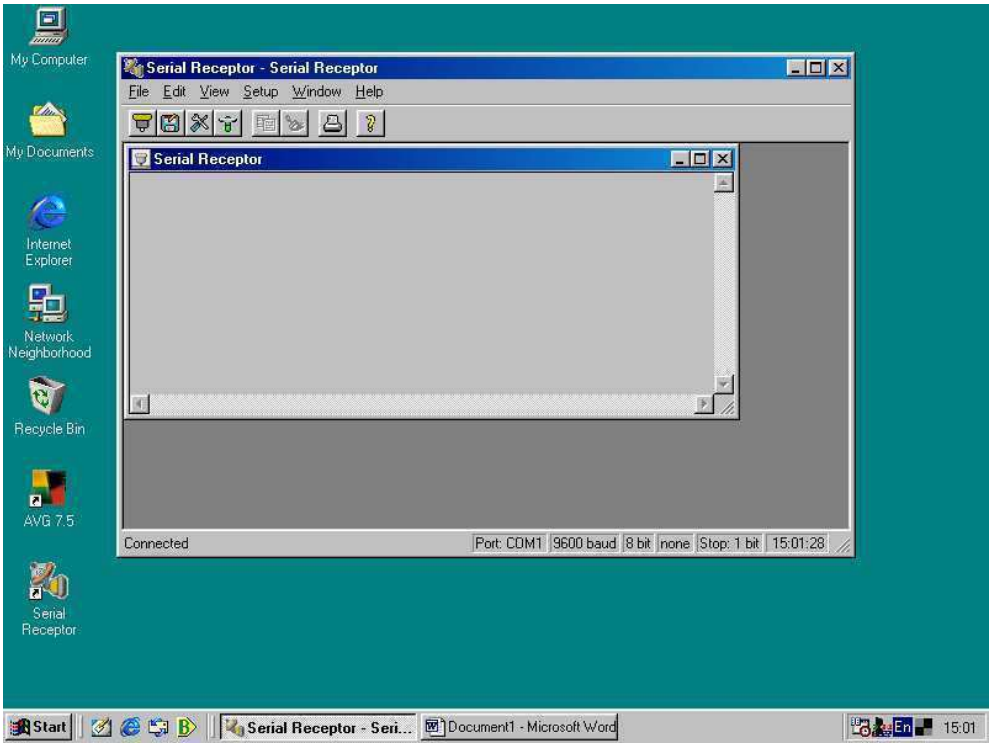
Διάγραμμα 3.11

Περιφερειακή μονάδα κάρτα I/O ISA BUS ZyXEL SP111AT (τοποθετημένη)



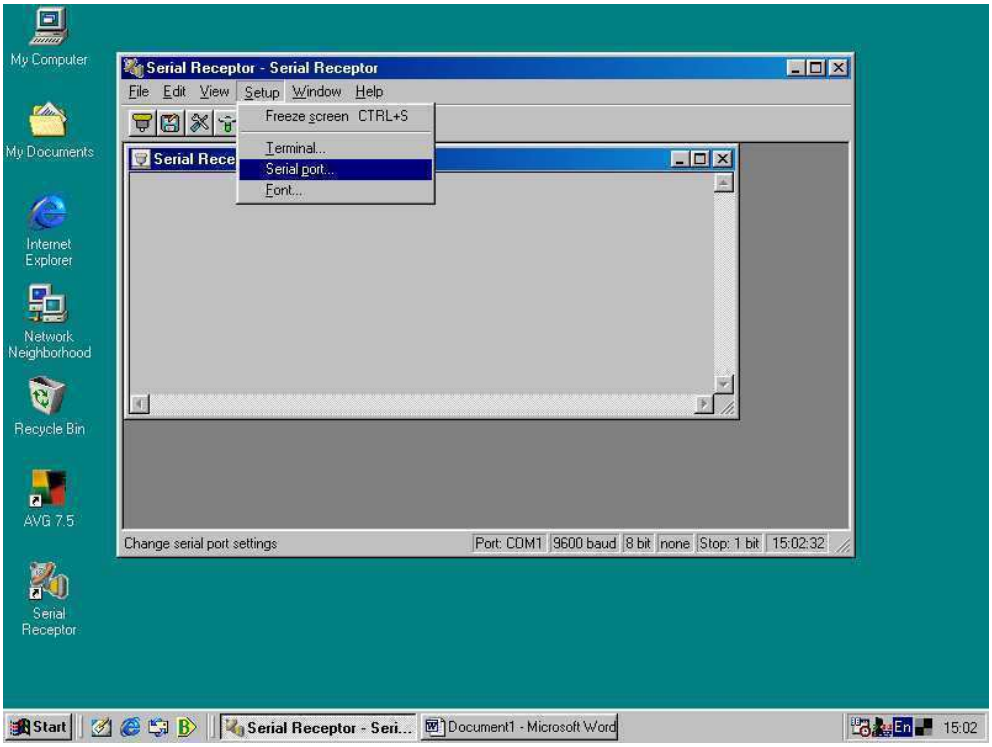
Διάγραμμα 3.12

Εκτύπωση οθόνης επιφάνειας εργασίας: συντόμευση του Serial Receptor



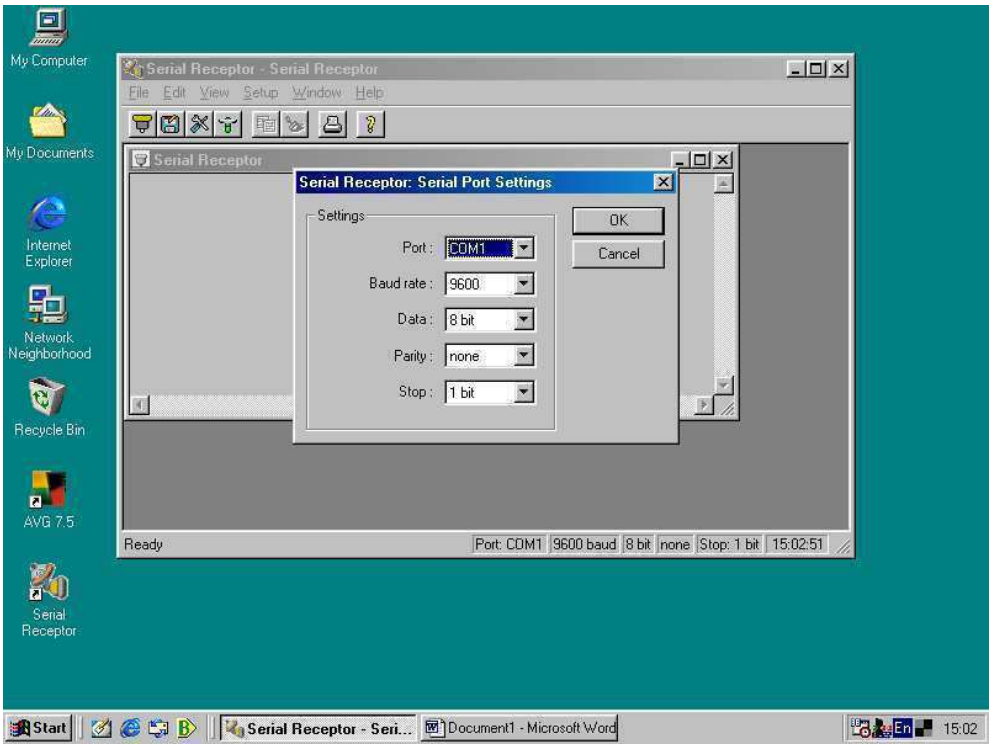
Διάγραμμα 3.13

Εκτύπωση οθόνης: "παράθυρο" Serial Receptor (παράθυρο παρακολούθησης θύρας COM1)



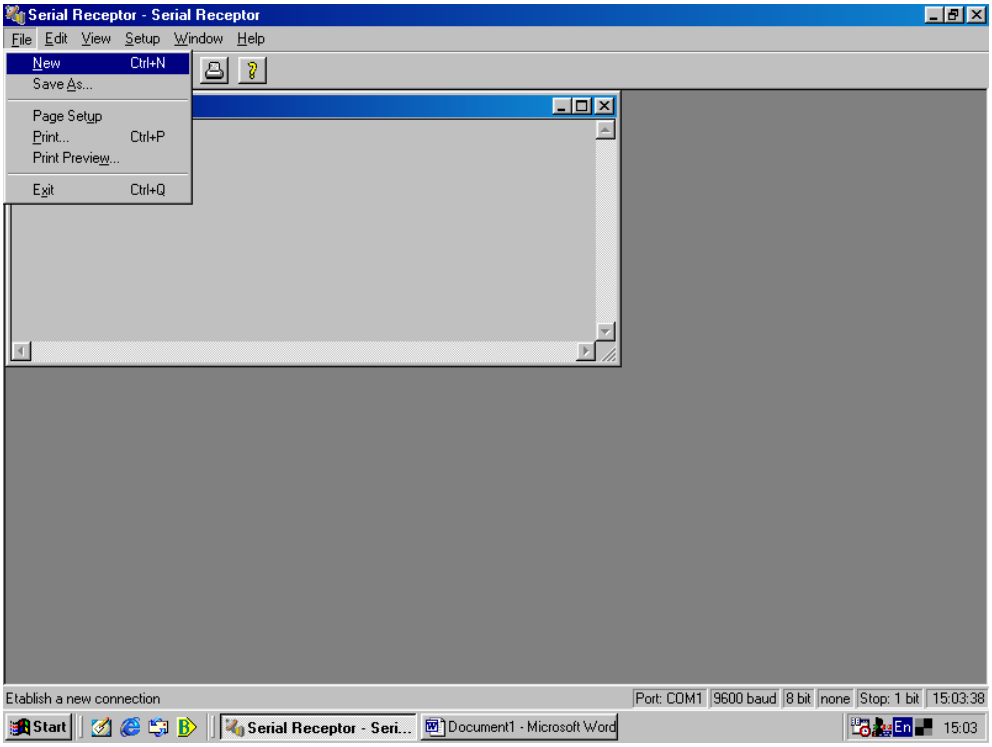
Διάγραμμα 3.14

Εκτύπωση οθόνης: ρύθμιση παραμέτρων θύρας COM1



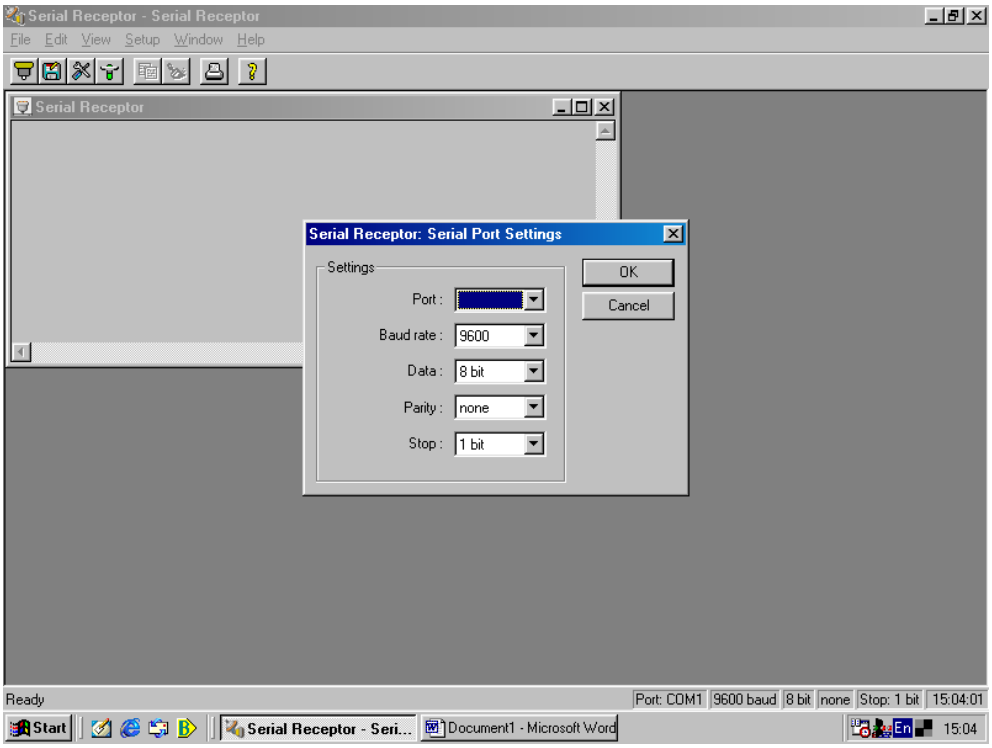
Διάγραμμα 3.15

Εκτύπωση οθόνης: επιλογές παραμέτρων θύρας COM1



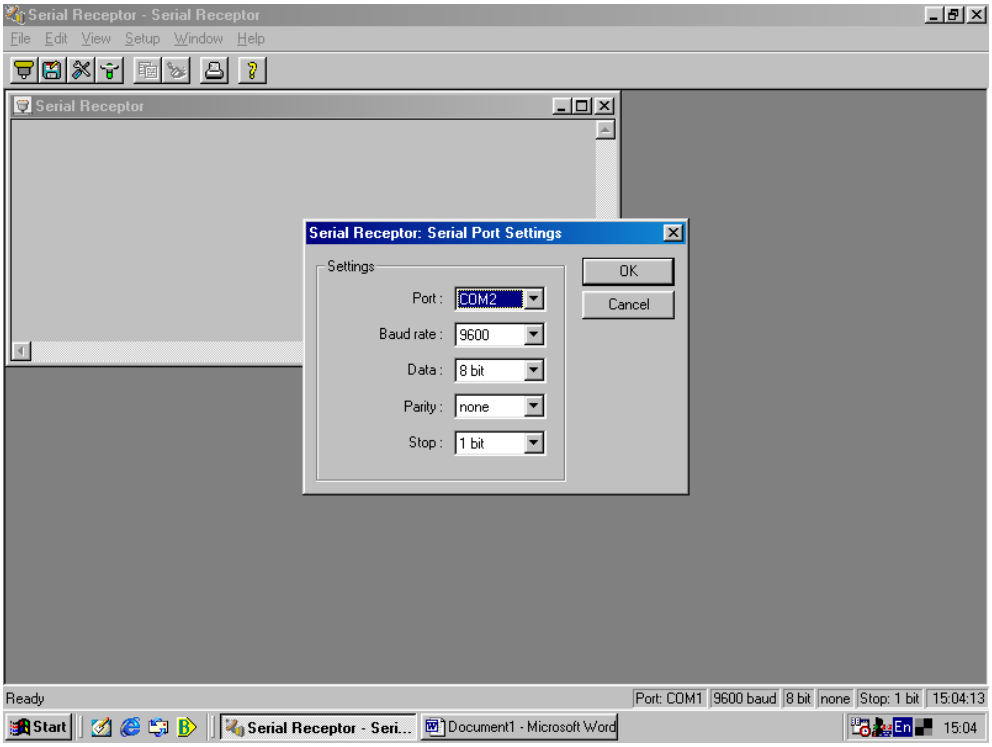
Διάγραμμα 3.16

Εκτύπωση οθόνης: έναρξη παρακολούθησης νέας θύρας



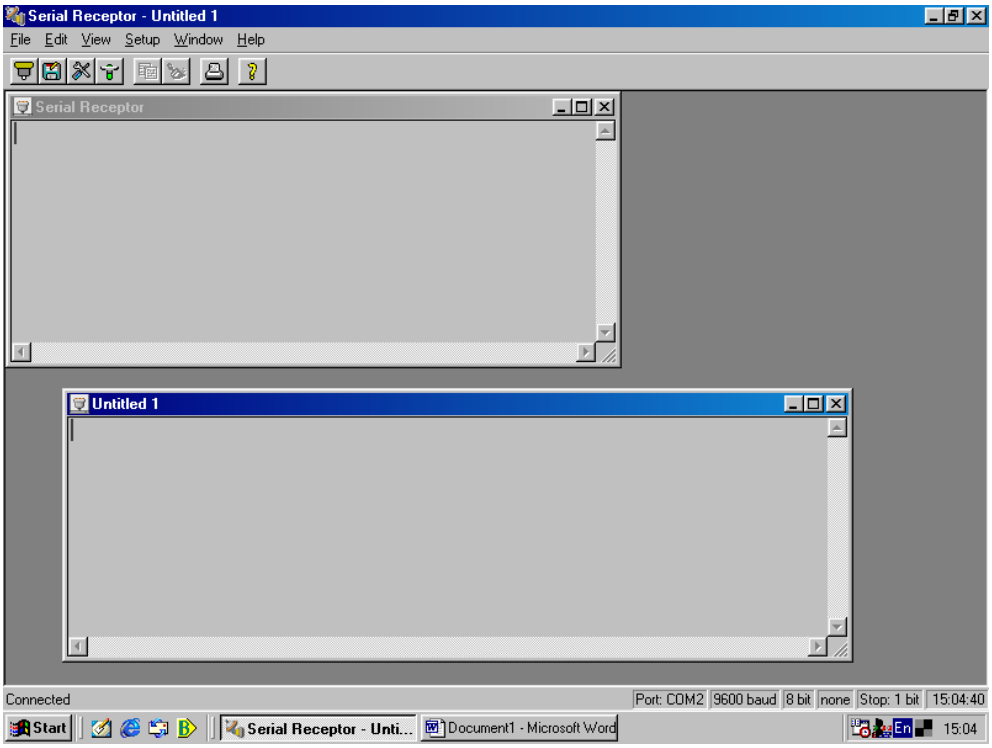
Διάγραμμα 3.17

Εκτύπωση οθόνης: επιλογές παραμέτρων νέας θύρας



Διάγραμμα 3.18

Εκτύπωση οθόνης: επιλογές παραμέτρων νέας θύρας, επιλογή COM2



Διάγραμμα 3.19

Εκτύπωση οθόνης: "παράθυρο" παρακολούθησης θύρας COM2 (Untitled1)

ΠΙΝΑΚΕΣ ΤΟΥ 3ου ΚΕΦΑΛΑΙΟΥ

ΣΗΜΑ		ΥΠΟΔΟΧΗ	
ΟΝΟΜΑ	ΚΩΔΙΚΟ ΟΝΟΜΑ	DB-25	DB-9
Common Ground	G	7	5
Transmitted Data	TxD	2	3
Received Data	RxD	3	2
Data Terminal Ready	DTR	20	4
Data Set Ready	DSR	6	6
Request To Send	RTS	4	7
Clear To Send	CTS	5	8
Carrier Detect	DCD	8	1
Ring Indicator	RI	22	9

Πίνακας 3.1

Τυποποίηση εννέα απαραίτητων σημάτων και αντίστοιχα pins σε βύσματα τύπου DB25 και DB9, αρσενικά DTE

ΣΗΜΑ		ΥΠΟΔΟΧΗ	
ΟΝΟΜΑ	ΚΩΔΙΚΟ ΟΝΟΜΑ	DB-25	DB-9
Common Ground	G	7	5
Transmitted Data	TxD	3	3
Received Data	RxD	2	2
Data Terminal Ready	DTR	20	4
Data Set Ready	DSR	6	6
Request To Send	RTS	4	7
Clear To Send	CTS	5	8
Carrier Detect	DCD	8	1
Ring Indicator	RI	22	9

Πίνακας 3.2

Τυποποίηση εννέα απαραίτητων σημάτων και αντίστοιχα pins σε βύσματα τύπου DB25 και DB9, θηλυκά DCE

IRQ	Μονάδες για Η/Υ AT, 386, 486, και Pentium
0	Χρονόμετρο Συστήματος
1	Πληκτρολόγιο
2	Συνδεδεμένο με τα IRQ 8 έως 15
3	Δεύτερη σειριακή θύρα (COM 2)(*)
4	Πρώτη σειριακή θύρα (COM 1)
5	Δεύτερη παράλληλη θύρα (LPT2) (*) ή κάρτα ήχου (**)
6	Ελεγκτής Δισκέτας
7	Πρώτη παράλληλη θύρα (LPT 1)
8	Real Time Clock
9	Συνδεδεμένο με το IRQ 2
10	Δεν έχει αποδοθεί
11	Δεν έχει αποδοθεί
12	Θύρα PS/2 για ποντίκι
13	Μονάδα Αριθμητικής Επεξεργασίας - Numerical Processing Unit (NPU)
14	Πρωτεύων ελεγκτής σκληρού δίσκου
15	Δευτερεύων ελεγκτής σκληρού δίσκου

(*) αν υπάρχει στη μητρική κάρτα

(**) αν είναι συνδεδεμένη και έχει εγκατασταθεί το λογισμικό της

Πίνακας 3.3

Συνήθη IRQ μονάδων Η/Υ τύπου PC συμβατού τεχνολογίας AT με επεξεργαστή Intel 80x86

ΚΕΦΑΛΑΙΟ 4
ΡΥΘΜΙΣΗ ΚΑΙ ΣΥΝΔΕΣΗ ΣΕΙΡΙΑΚΗΣ ΘΥΡΑΣ Η/Υ ΚΑΤΩ ΑΠΟ ΤΟ Λ/Σ LINUX

4.1 Εισαγωγή

Το λογισμικό ρύθμισης, αποστολής και λήψης δεδομένων για σειριακή θύρα, όπως αναπτύχθηκε για το Λ/Σ UNIX System V από τον Χαρβάτη (1990) συντάχθηκε σε γλώσσα προγραμματισμού C και καλεί υπορουτίνες βιβλιοθήκης του Λ/Σ UNIX System V. Η γλώσσα αυτή επιλέχθηκε διότι έχει τη δυνατότητα να πραγματοποιεί εύκολα κλήσεις συστήματος και να καλεί συναρτήσεις βιβλιοθήκης του Λ/Σ, πράγμα που είναι απαραίτητο για τον τελικό έλεγχο των περιφερειακών (όπως δηλαδή είναι η σειριακή θύρα). Το λογισμικό αυτό διακρίθηκε σε τρεις ενότητες υποπρογραμμάτων: (α) τα υποπρογράμματα ρύθμισης της σειριακής θύρας, (β) τα υποπρογράμματα επικοινωνίας με τις μονάδες επικοινωνίας CANBERRA και (γ) τα γενικά υποπρογράμματα. Το σύνολο των υποπρογραμμάτων συγκροτήθηκε σε κατάλληλη βιβλιοθήκη, ώστε να διευκολυνθεί η χρήση τους και η κλήση τους από προγράμματα γραμμένα σε γλώσσα FORTRAN. Στο Κεφάλαιο αυτό παρουσιάζονται διαδοχικά ένα προς ένα όλα τα πιοπάνω υποπρογράμματα, όπως μετατράπηκαν προκειμένου να λειτουργούν κάτω από το Λ/Σ LINUX με Η/Υ τεχνολογίας PC συμβατού, με τον ίδιο ακριβώς τρόπο που λειτουργούσαν κάτω από το Λ/Σ UNIX System V με Η/Υ HP 9000. Φυσικά μετά τη μετατροπή οι καλούμενες συναρτήσεις βιβλιοθήκης δεν είναι πλέον προτύπου System V, αλλά προτύπου POSIX, σύμφωνα και με τα όσα αναφέρονται στο 3^{ο} Κεφάλαιο. Οι διαφορές που προέκυψαν στα υποπρογράμματα, μετά τη μετατροπή, δηλαδή οι προσθέσεις, αφαιρέσεις και αλλαγές σημαίνονται με κατάλληλη σκίαση [ ]. Επισημαίνεται ότι, σε όσα αφορούν στα υποπρογράμματα ρύθμισης σειριακής θύρας, ως θύρα εννοείται η θύρα του Η/Υ και όχι η όποια θύρα της όποιας μονάδας επικοινωνίας.

4.2 Λ/Σ LINUX και περιφερειακές συσκευές

Όπως σημειώνεται και από τον Χαρβάτη (1990), κάτω από το UNIX (και βεβαίως ισοδύναμα κάτω από το LINUX), η απευθείας προσπέλαση του υλικού (hardware) του συστήματος είναι σχεδόν αδύνατη για τον χρήστη - προγραμματιστή. Μεταξύ τους παρεμβάλλεται το επίπεδο του πυρήνα του συστήματος. Το σχήμα του Διαγράμματος 4.1 είναι μία απλοποιημένη παράσταση της δομής του πυρήνα του

συστήματος, όπου διακρίνονται τα δύο βασικά συστατικά του, το process control subsystem που χειρίζεται τα προγράμματα που "τρέχουν" στον Η/Υ (processes) και το file subsystem που χειρίζεται τα αρχεία αλλά και τις περιφερειακές συσκευές του Η/Υ. Στο ίδιο Διάγραμμα διακρίνεται και ένα ακόμη στοιχείο που υπάγεται στο file subsystem: οι device drivers. Για κάθε περιφερειακή συσκευή (device) του Η/Υ υπάρχουν στον πυρήνα του Λ/Σ κάποια υποπρογράμματα που ελέγχουν άμεσα τη συσκευή. Τα υποπρογράμματα αυτά αποτελούν τον οδηγό της συσκευής (device driver).

Οι διάφορες συσκευές (και οι αντίστοιχοι drivers) χωρίζονται σε :

character devices

block devices

Character devices είναι συσκευές, όπως τα τερματικά, οι εκτυπωτές και γενικά οι ψηφιακές συσκευές , στις οποίες η μετάδοση των πληροφοριών γίνεται χαρακτήρα - χαρακτήρα.

Block devices είναι συσκευές, όπως ο μαγνητικός δίσκος, όπου συμφέρει, για αύξηση της ταχύτητας, η μετάδοση των πληροφοριών να γίνεται κατά ομάδες (blocks) χαρακτήρων σταθερού μήκους.

Η σύνδεση και η επικοινωνία των προγραμμάτων του χρήστη με τον πυρήνα του Λ/Σ γίνεται μέσω των κλήσεων συστήματος (στα αγγλικά system calls). Οι κλήσεις αυτές, που στο πρόγραμμα του χρήστη υλοποιούνται με απλές κλήσεις υποπρογραμμάτων του Λ/Σ, χωρίζονται σε δύο βασικές κατηγορίες : (α) Σε όσες αλληλεπιδρούν με το process control subsystem και αφορούν χειρισμούς των processes του Η/Υ, και (β) Σε όσες αλληλεπιδρούν με το file subsystem και αφορούν χειρισμούς αρχείων, αλλά και χειρισμούς των περιφερειακών συσκευών του Η/Υ.

Στα πλαίσια της παρούσης ΔΕ ενδιαφέρει η εξυπηρέτηση των αναγκών του χρήστη από το file subsystem, και επομένως απαιτούνται χειρισμοί αρχείων όσο και περιφερειακών συσκευών. Αυτοί οι χειρισμοί γίνονται με ένα ομοιόμορφο τρόπο και με τις ίδιες κλήσεις συστήματος. Για να μπορεί να γίνει αυτό, όλες οι συσκευές του Η/Υ θεωρούνται από το UNIX ή το LINUX ότι συμπεριφέρονται ως αρχεία. Τα αρχεία αυτά ονομάζονται ειδικά αρχεία (στα αγγλικά special files). Κάθε χειρισμός πάνω σε ένα τέτοιο αρχείο περνά μέσω του file subsystem και του device driver πάνω στην ίδια τη συσκευή. Με αυτόν τον μηχανισμό, οι χειρισμοί και ο έλεγχος του

υλικού των περιφερειακών συσκευών του Η/Υ προσομοιάζονται στο επίπεδο του χρήστη ως χειρισμοί αρχείων. Πολλές φορές μάλιστα το πρόγραμμα του χρήστη δεν γνωρίζει - ούτε χρειάζεται να γνωρίζει - αν χειρίζεται απλά αρχεία ή περιφερειακές συσκευές.

Οι κυριότερες κλήσεις συστήματος, που επιδρούν πάνω στα ειδικά αρχεία, και η περιγραφή της λειτουργίας τους στο βαθμό που δεν εξαρτάται από κάποιο ιδιαίτερο χαρακτηριστικό της συσκευής που εξυπηρετούν περιγράφονται σε εγχειρίδια UNIX / LINUX, όπως π.χ. το Hewlett Packard (1988, Section 2).

4.3 Κλήσεις συστήματος για τον έλεγχο σειριακής θύρας RS-232

4.3.1 Γενικά

Οι κλήσεις συστήματος είναι, όπως σημειώθηκε προηγούμενα, ο τρόπος με τον οποίο το περιβάλλον εφαρμογών του χρήστη μπορεί να επικοινωνήσει με τις υπηρεσίες που μπορεί να παρέχει ο πυρήνας (kernel) του Λ/Σ. Οι κλήσεις συστήματος πρέπει να γίνονται με συγκεκριμένους κανόνες και διαδικασίες ώστε να επιτυγχάνεται η μετάβαση από το επίπεδο του χρήστη στο επίπεδο του πυρήνα (βλ. και το σχήμα του Διαγράμματος 4.1).Οι κλήσεις συστήματος σε ένα τυπικό Λ/Σ LINUX υπολογίζονται σήμερα (2008) σε περίπου 320. Ο Χαρβάτης (1990) προτείνει ότι για τον έλεγχο και την επικοινωνία με περιφερειακές συσκευές τύπου character devices, όπως η θύρα RS-232 μπορούν να χρησιμοποιούνται οι ακόλουθες κλήσεις συστήματος:

open, read, write, close, lockf, ioctl

4.3.2 Κλήση συστήματος open

Σύνταξη: **int open (*pathname, flags)**

const_char *pathname;

int flags;

Περιγραφή: Η κλήση αυτή ενεργοποιεί μια σύνδεση μεταξύ μιας περιφερειακής συσκευής και του καλούντος προγράμματος και δίνει αρχικές τιμές στις μεταβλητές του οδηγού της συσκευής. Όπως και για τα απλά αρχεία, επιστρέφει έναν κωδικό αριθμό θετικό ακέραιο (τον λεγόμενο file descriptor). Ο κωδικός αυτός αριθμός, χαρακτηρίζει εφεξής το ειδικό αρχείο και μέσω αυτού πραγματοποιείται κάθε περαιτέρω επικοινωνία με τη συσκευή. Στην εδώ περίπτωση, η κλήση **open**

αποκαθιστά τη σύνδεση με την θύρα RS-232 και επιστρέφει τον file descriptor που χρειάζεται σε όλες τις υπόλοιπες κλήσεις. Αν αποτύχει επιστρέφει **-1**.

***pathname** είναι το όνομα του ειδικού αρχείου που αντιστοιχεί στην RS-232. Τα ειδικά αρχεία που αντιστοιχούν στις περιφερειακές συσκευές του Η/Υ βρίσκονται στο directory /dev. Τα ονόματά τους καθορίζονται από τον system manager, ο οποίος κάνει και την σύνδεση των αρχείων αυτών με τις περιφερειακές συσκευές. Σε ένα Η/Υ τύπου PC συμβατό, τα ονόματα των ειδικών αρχείων που αντιστοιχούν στις θύρες επικοινωνίας RS-232 έχουν τη μορφή ttySi (όπου i ένας ακέραιος αριθμός από 0 έως 75). Η θύρα RS-232 του Η/Υ PC συμβατού που χρησιμοποιείται για την σύνδεση των συσκευών στην παρούσα ΔΕ είναι η /dev/ttyS0, η οποία είναι η πρώτη (από τις δύο διαθέσιμες) σειριακή θύρα στην μητρική κάρτα του Η/Υ που χρησιμοποιείται (η οποία αν το Λ/Σ ήταν DOS θα αποδίδονταν ως COM1). Η παράμετρος **flags** πρέπει να έχει την τιμή **O_RDWR** εφόσον πρόκειται για αμφίδρομη επικοινωνία που θα επιτρέψει ανάγνωση από και εγγραφή προς τη σειριακή θύρα. Περισσότερες πληροφορίες σχετικά με τη λειτουργία της κλήσης συστήματος **open** μπορεί κανείς να βρει δίνοντας σε Λ/Σ LINUX:

$ man –S 2 open

$ man types.h

$ man stat.h

$ man fcntl.h

4.3.3 Κλήση συστήματος read

Σύνταξη: **int read (fd,* buf, count)**

int fildes;

void *buf;

size_t count;

Περιγραφή: Κλήση που διαβάζει πληροφορίες που έρχονται από την περιφερειακή συσκευή. Στην εδώ περίπτωση επιχειρεί να διαβάσει **count** χαρακτήρες από την RS-

232 και τους τοποθετεί στη θέση ***buf**. Επιστρέφει τον αριθμό των χαρακτήρων που διαβάστηκαν τελικά, ή **-1**, αν αποτύχει.

fd είναι το file descriptor (στην εδώ περίπτωση είναι στην ουσία η περιφερειακή συσκευή /dev/ttyS0). Στην περίπτωση λειτουργίας με τον "κανονικό τρόπο"[1], η **read** επιστρέφει μόνο όταν ολοκληρωθεί η γραμμή. Στην περίπτωση λειτουργίας με τον "μη κανονικό τρόπο", το πότε θα επιστρέψει η **read** καθορίζεται από τους ειδικούς χαρακτήρες **MIN** και **TIME**, όπως αναφέρεται σε επόμενη παράγραφο. Περισσότερες πληροφορίες σχετικά με τη λειτουργία της κλήσης συστήματος **read** μπορεί κανείς να βρει δίνοντας σε Λ/Σ LINUX:

$ man –S 2 read

$ man unistd.h

4.3.4 Κλήση συστήματος write

Σύνταξη: **int write (fd, *buf, count)**

int fd;

const void *buf;

size_t count;

Περιγραφή: Κλήση που αποστέλλει πληροφορίες από τον Η/Υ προς την περιφερειακή συσκευή. Στην εδώ περίπτωση επιχειρεί να στείλει **count** χαρακτήρες στην RS-232. Επιστρέφει τον αριθμό των χαρακτήρων που εστάλησαν ή **–1** αν αποτύχει.

fd είναι το file descriptor (στην εδώ περίπτωση είναι στην ουσία η περιφερειακή μονάδα /dev/ttyS0). Περισσότερες πληροφορίες σχετικά με τη λειτουργία της κλήσης συστήματος **write** μπορεί κανείς να βρει δίνοντας σε Λ/Σ LINUX:

$ man –S 2 write

$ man unistd.h

[1] βλ. σε επόμενη παράγραφο

4.3.5 Κλήση συστήματος close

Σύνταξη: **int close(fd)**

int fd;

Περιγραφή: Με την κλήση αυτή η σύνδεση με τη περιφερειακή συσκευή τερματίζεται. Στην εδώ περίπτωση τερματίζεται τη σύνδεση με την RS-232. Αν η κλήση **close** επιτύχει, επιστρέφει **0** , αλλιώς **-1**.

fd είναι το file descriptor (στην εδώ περίπτωση είναι στην ουσία η περιφερειακή μονάδα /dev/ttyS0). Περισσότερες πληροφορίες σχετικά με τη λειτουργία της κλήσης συστήματος **close** μπορεί κανείς να βρει δίνοντας σε Λ/Σ LINUX:

$ man –S 2 close

$ man unistd.h

4.3.6 Κλήση συστήματος lockf

Σύνταξη: **int lockf (fd, cmd, len)**

int fd, cmd;

off_t len;

Περιγραφή: Με την κλήση αυτή, το καλούν πρόγραμμα (process) μπορεί να ελέγξει αν το ειδικό αρχείο είναι "κλειδωμένο" δηλαδή αν η περιφερειακή συσκευή χρησιμοποιείται από κάποια άλλη process. Στην περίπτωση που η συσκευή είναι ελεύθερη, η lockf, μπορεί να την "κλειδώσει" για λογαριασμό της process που την κάλεσε. Οι παράμετροι **function** και **size** καθορίζουν την λειτουργία της **lockf**. Αν **cmd = F_TLOCK** και **len = 0**, η **lockf** επιχειρεί να κλειδώσει το ειδικό αρχείο της RS-232. Σε περίπτωση επιτυχίας επιστρέφει **0**. Αν αποτύχει επιστρέφει **-1** (αυτό θα συμβεί αν κάποια άλλη process χρησιμοποιεί την RS-232 και έχει ήδη κλειδώσει το ειδικό αρχείο).

fd είναι το file descriptor (στην εδώ περίπτωση είναι στην ουσία η περιφερειακή μονάδα /dev/ttyS0) Περισσότερες πληροφορίες σχετικά με τη λειτουργία της κλήσης συστήματος **lockf** μπορεί κανείς να βρει δίνοντας σε Λ/Σ LINUX:

$ man lockf

$ man unistd.h

4.3.7 Κλήση συστήματος ioctl

Περιγραφή: Κλήση που καθορίζει διάφορα χαρακτηριστικά της περιφερειακής συσκευής. Είναι η μόνη κλήση που δεν είναι διαθέσιμη για απλά αρχεία. Η κλήση αυτή διατίθεται τόσο για Λ/Σ UNIX / System V, όσο και για Λ/Σ LINUX / POSIX. Παρόλαυτά, η κλήση αυτή δεν χρησιμοποιείται, στην προγραμματιστική υλοποίηση της παρούσας ΔΕ, διότι, όπως θα φανεί στη συνέχεια, διατίθενται αντί για αυτήν άλλα προγραμματιστικά εργαλεία με πιο εύκολη χρήση.

4.4 Συναρτήσεις βιβλιοθήκης POSIX

4.4.1 Γενικά

Οι συναρτήσεις βιβλιοθήκης κατά το πρότυπο POSIX, είναι υποπρογράμματα ανεξάρτητα της γλώσσας προγραμματισμού που τα καλεί. Παρόλαυτά, η κλήση τους μπορεί να γίνεται με το συμβατικό τρόπο που ορίζεται για τη γλώσσα προγραμματισμού C. Κατ' αυτόν τον τρόπο οι συναρτήσεις βιβλιοθήκης POSIX, μπορούν να θεωρηθούν στα πλαίσια αυτής της ΔΕ, ως επιπλέον και συμπληρωματικές συναρτήσεις της γλώσσας προγραμματισμού C. Οι συναρτήσεις βιβλιοθήκης POSIX, αναπτύχθηκαν την ίδια χρονική περίοδο με την γλώσσα προγραμματισμού C, κατά το πρότυπο ANSI. Κάποιες από τις συνηθέστερα χρησιμοποιούμενες συναρτήσεις βιβλιοθήκης POSIX, είναι οι:

cpio.h, dirent.h, fcntl.h, grp.h, pwd.h, sys/ipc.h, sys/msg.h, sys/sem.h, sys/stat.h, sys/time.h, sys/types.h, sys/utsname.h, sys/wait.h, tar.h, termios.h, unistd.h, utime.h

Από αυτές στα πλαίσια της παρούσης ΔΕ, σύμφωνα με τον προγραμματισμό που υλοποίησε και ο Χαρβάτης (1990), χρησιμοποιούνται οι:

fcntl.h, sys/time.h, termios.h, unistd.h

οι οποίες διεκπεραιώνουν, σχηματικά, τα πιο κάτω καθήκοντα:

<fcntl.h> Διαχείριση αρχείων ("άνοιγμα", "κλείδωμα", κ.ά.)

<sys/time.h> Υπορουτίνες ώρας ημερομηνίας

<termios.h> γενικευμένος κώδικας επικοινωνίας για τον έλεγχο εισόδου εξόδου σε ασύγχρονες επικοινωνίες.

<unistd.h> διάφορες υπορουτίνες.

Από τις πιοπάνω 4 συναρτήσεις βιβλιοθήκης POSIX, σημαντικότερη για τους σκοπούς της ΔΕ είναι η **termios.h.** Με την βοήθεια της **termios** μπορούν να ελέγχονται οι θύρες RS-232 που χρησιμοποιούνται για τη σύνδεση τερματικών, εκτυπωτών και λοιπών ψηφιακών συσκευών με τον Η/Υ κάτω από το Λ/Σ LINUX.

4.4.2 Έλεγχος σειριακής θύρας RS-232 μέσω της συνάρτησης termios.h

Κατά το παλαιότερο πρότυπο System V, η συνάρτηση βιβλιοθήκης που διεκπεραιώνει ασύγχρονες επικοινωνίες με περιφερειακά ανεξάρτητα από το χρησιμοποιούμενο υλικό (εκτός από τις δικτυακές συνδέσεις), κάτω από το Λ/Σ UNIX είναι η **termio.h**, η οποία και χρησιμοποιείται στην υλοποίηση του Χαρβάτη (1990). Κατά το τρέχον πρότυπο POSIX, η συνάρτηση βιβλιοθήκης που διεκπεραιώνει ασύγχρονες επικοινωνίες με περιφερειακά ανεξάρτητα από το χρησιμοποιούμενο υλικό, κάτω από το Λ/Σ LINUX είναι η **termios.h.** Οι βασικές λειτουργίες των δύο συναρτήσεων είναι παρόμοιες, παρόλαυτά η κυριότερη διαφορά τους είναι ότι η **termio.h** / System V χρησιμοποιεί και την κλήση συστήματος **ioctl**, για τη ρύθμιση κάποιων από τις παραμέτρους επικοινωνίας, πράγμα που δεν γίνεται από την **termios.h** / POSIX, διότι η κλήση συστήματος **ioctl** δεν είναι γενικά προτυποποιημένη. Στη θέση των κλήσεων **ioctl** η **termios.h** χρησιμοποιεί νέο προτυποποιημένο κώδικα.

Η συνάρτηση **termios.h** ρυθμίζεται παραμετρικά, ανάλογα με τα καθήκοντα επικοινωνίας που πρέπει να διεκπεραιώσει. Από αυτή την άποψη η **termios.h** λειτουργεί ως ρυθμιζόμενος οδηγός ασύγχρονης επικοινωνίας (ή όπως λέγεται στα αγγλικά terminal[2] driver). Η **termios.h** διαφοροποιείται σημαντικά από τους drivers λοιπών συσκευών (π.χ. οθόνη, ήχος), ως προς την ευελιξία που παρέχει στο χρήστη για την τροποποίηση των λειτουργικών χαρακτηριστικών της, ώστε να είναι εφικτή η εξυπηρέτηση κάθε τύπου περιφερειακής συσκευής ασύγχρονης επικοινωνίας.

Όταν ο χρήστης ανοίγει το αρχείο που αντιστοιχεί στην περιφερειακή συσκευή ασύγχρονης επικοινωνίας (στην εδώ περίπτωση η σειριακή θύρα RS-232), η

[2] όπου terminal κάθε περιφερειακή συσκευή ασύγχρονης επικοινωνίας μέσω της οποίας ο χρήστης επικοινωνεί με τον υπολογιστή

termios.h περιμένει τη σύνδεση της συσκευής. Η συσκευή μπορεί να εργάζεται "full-duplex" ή "half-duplex". Κατά τη λειτουργία "full-duplex", ταυτόχρονα δεδομένα έρχονται από τη συσκευή στον Η/Υ, και αποστέλλονται από τον Η/Υ στη συσκευή. Απώλεια δεδομένων παρουσιάζεται μόνο όταν οι σχετικοί καταχωρητές εξαντλήσουν τη χωρητικότητά τους. Κατά τη λειτουργία "half-duplex", τα δεδομένα κινούνται μόνο προς μία κατεύθυνση, δηλαδή είτε από τη συσκευή προς τον Η/Υ, είτε από τον Η/Υ προς τη συσκευή. Στην περίπτωση της σειριακής θύρας RS-232, εννοείται προφανώς λειτουργία "half-duplex".

Όταν υπάρχει είσοδος δεδομένων από τη σειριακή θύρα προς τον Η/Υ, η επεξεργασία τους από την **termios.h** μπορεί να γίνεται κατά τον λεγόμενο "κανονικό τρόπο" (στα αγγλικά canonical mode) ή κατά τον λεγόμενο "μη κανονικό τρόπο" (non-canonical mode), ή/και με διάφορους ειδικότερους τρόπους που ρυθμίζονται με ειδικούς διακόπτες στους οποίους θα γίνει αναφορά αργότερα.

Κατά το canonical mode για την επεξεργασία της εισόδου, τα δεδομένα εισόδου εξετάζονται σε ομάδες που ονομάζονται "γραμμές". Κάθε γραμμή ολοκληρώνεται, όταν συμπληρωθεί το μήκος της (ο καθορισμένος αριθμός χαρακτήρων που μπορεί να περιέχει), ή όταν ληφθεί ο ειδικός χαρακτήρας carriage return (CR), ή ο ειδικός χαρακτήρας new-line (NL), ή ο ειδικός χαρακτήρας end-of-line (EOL), ή ο ειδικός χαρακτήρας end-of-file (EOF). Αυτό σημαίνει ότι μία κλήση **read** δεν επιστρέφει παραμόνο όταν ολοκληρωθεί μία γραμμή. Έτσι, η κλήση **read** κατά το canonical mode, επιστρέφει οπωσδήποτε, όταν ολοκληρωθεί μία γραμμή, ακόμα και αν ο αριθμός των χαρακτήρων που ζητούνται να διαβασθούν είναι περισσότεροι από το μήκος μιας γραμμής. Αν οι χαρακτήρες αυτοί πράγματι υπάρχουν στον καταχωρητή εισόδου, χάνονται. Επιπλέον μία κλήση **read** δεν είναι υποχρεωτικό να διαβάσει μία πλήρη γραμμή, μπορεί να διαβάσει και ένα μόνο τμήμα της με μήκος μικρότερο από το μήκος της γραμμής. Τα δεδομένα που έχουν διαβασθεί από μία γραμμή, κατά το canonical mode υπάγονται και στην επεξεργασία "erase and kill". Αυτή συμβαίνει όταν λαμβάνονται οι ειδικοί χαρακτήρες ERASE και KILL, και η λήψη των δεδομένων της γραμμής δεν έχει ολοκληρωθεί με τη λήψη και των ειδικών χαρακτήρων CR, NL, EOL, ή EOF. Σε τέτοια περίπτωση, ο ειδικός χαρακτήρας ERASE διαγράφει τον αμέσως προηγούμενο χαρακτήρα στη γραμμή, αν αυτός υπάρχει, ενώ ο ειδικός χαρακτήρας KILL διαγράφει την τρέχουσα γραμμή. Οι ειδικοί χαρακτήρες ERASE και KILL δεν έχουν καμία επίδραση αν η τρέχουσα γραμμή είναι

άδεια. Συνοψίζοντας κατά το canonical mode η επεξεργασία της εισόδου γίνεται ως εξής:

- Αναγνωρίζονται οι ειδικοί χαρακτήρες CR, NL, EOL, EOF, πράγμα που επιτρέπει την ομαδοποίηση των εισερχομένων χαρακτήρων κατά γραμμές.
- Διαγράφεται ο τελευταίος χαρακτήρας αν έρθει ειδικός χαρακτήρας ERASE.
- Διαγράφεται η τελευταία γραμμή αν έρθει ειδικός χαρακτήρας KILL.

Κατά το non-canonical mode για την επεξεργασία της εισόδου, τα δεδομένα εισόδου δεν εξετάζονται κατά γραμμές και δεν υπάρχει η επεξεργασία "erase and kill". Η επεξεργασία γίνεται με τρόπο που ορίζουν οι τιμές των διακοπτών MIN και TIME, που ανήκουν στην παράμετρο c_cc array[3] της συνάρτησης βιβλιοθήκης **termios.h**. Ισοδύναμα η συμπεριφορά της κλήσης **read** καθορίζεται ακριβώς από αυτούς τους διακόπτες. Αν, όταν κληθεί η **read**, στον καταχωρητή εισόδου υπάρχουν τουλάχιστον MIN χαρακτήρες, η **read** επιστρέφει αμέσως διαβάζοντας τους χαρακτήρες αυτούς. Στην αντίθετη περίπτωση, αν δηλαδή στον καταχωρητή εισόδου υπάρχουν λιγότεροι από MIN χαρακτήρες, η **read** επιστρέφει μόνο αν έρθουν αρκετοί νέοι χαρακτήρες προς την RS232 ώστε να συμπληρωθεί ο αριθμός MIN ή αν περάσουν TIME δέκατα δευτερολέπτου.

Γίνεται αντιληπτό ότι στην περίπτωση σειριακής ανταλλαγής δεδομένων με μονάδες επικοινωνίας CANBERRA, όπως απαιτείται στα πλαίσια της παρούσης ΔΕ, η σειριακή επικοινωνία για την ανάγνωση των καταμετρητών πρέπει να γίνεται σε non-canonical mode, ώστε τα δεδομένα να έρχονται στον Η/Υ χαρακτήρα προς χαρακτήρα και όχι κατά γραμμές. Επομένως πρακτικό ενδιαφέρον παρουσιάζουν οι εξής περιπτώσεις :

MIN = 1 , TIME = 0 : η **read** δεν επιστρέφει παρά μόνο όταν υπάρξει ένας τουλάχιστον χαρακτήρας.

MIN = 0 , TIME = 0 : η **read** επιστρέφει αμέσως ανεξάρτητα από το αν υπάρχει χαρακτήρας ή όχι.

Προκειμένου να αρχίσει η επικοινωνία μιας σειριακής περιφερειακής συσκευής με τον Η/Υ μέσω της θύρας RS-232 πρέπει να καθορισθούν τα λειτουργικά χαρακτηριστικά του οδηγού ασύγχρονης επικοινωνίας (terminal driver). Αυτό

[3] βλ. και στη συνέχεια για τη σύνταξη της δομής **termios.h**

πρακτικά επιτυγχάνεται με τον καθορισμό των τιμών μιας ομάδας διακοπτών της **termios.h**,που συγκροτούν μια προκαθορισμένη δομή (structure) **termios** στο Λ/Σ. Δηλαδή, η δομή **termios** είναι ένας τύπος μεταβλητής σχεδιασμένος ειδικά για να κρατάει τις ρυθμίσεις της σειριακής θύρας RS-232 και τις παραμέτρους που καθορίζουν τον τρόπο λειτουργίας του terminal driver. Ο ορισμός της δομής **termios** στηρίζεται στη συνάρτηση βιβλιοθήκης **termios.h** / POSIX. Θα πρέπει να είναι:

Σύνταξη (βλ. και **man termios.h**, **man termios**)

```
struct termios {
tcflag_t c_iflag;       /* input modes */
tcflag_t c_oflag;       /* output modes */
tcflag_t c_cflag;       /* control modes */
tcflag_t c_lflag;       /* local modes */
cc_t    c_cc[NCCS];     /* control chars */
};
```

Αυτή η σύνταξη διαφέρει ελαφρά από τη σύνταξη της παλαιότερης δομής **termio** όπως παρουσιάζεται από τον Χαρβάτη (1990) και αφορά στην συνάρτηση βιβλιοθήκης **termio.h** / System V. Η χρήση της δομής **termio**, είχε τότε ως εξής, σύμφωνα και με Hewlett Packard (1988, Section 7) – βλ. και το κείμενο TERMIO(7) στο Παράρτημα (η διαφορά σημαίνεται με σκίαση και αναφέρεται στην παράμετρο c_line, η οποία δεν υπάρχει στην χρήση της δομής **termios**):

```
struct termio {
        unsigned short      c_iflag;      /* input modes */
        unsigned short      c_oflag;      /* output modes */
        unsigned short      c_cflag;      /* control modes */
        unsigned short      c_lflag;      /* local modes */
        char                c_line;       /* line discipline*/
        unsigned char       c_cc[NCC];    /* control chars */
};
```

Περιγραφή (βλ. και **man termios.h, man termios**)

Οι παράμετροι c_iflag, c_oflag, c_cflag, c_lflag, c_cc[NCCS] ρυθμίζουν τον τρόπο λειτουργίας του terminal driver ως εξής:

c_iflag : Η παράμετρος c_iflag ρυθμίζει τον τρόπο λειτουργίας του terminal driver όσον αφορά τους χαρακτήρες που λαμβάνονται από τη θύρα RS-232. Υπάρχουν αρκετές δυνατές τιμές για την παράμετρο αυτή, οι οποίες ονομαστικά είναι οι IGNBRK, BRKINT, IGNPAR, PARMRK, INPCK, ISTRIP, INCLR, IGNCR, IUCLC (non-POSIX), IXON, IXANY, IXOFF και IMAXBEL (non-POSIX). Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990) χρησιμοποιείται ως απαραίτητη η τιμή IGNPAR σύμφωνα με την οποία:

> **IGNPAR :** Οι χαρακτήρες με parity errors ή και άλλα λάθη μετάδοσης αγνοούνται (δεν τοποθετούνται στον καταχωρητή εισόδου).

c_oflag : Η παράμετρος c_oflag ρυθμίζει τον τρόπο λειτουργίας του terminal driver όσον αναφορά τους χαρακτήρες που αποστέλλονται στην θύρα RS-232. Υπάρχουν αρκετές δυνατές τιμές για την παράμετρο αυτή, οι οποίες ονομαστικά είναι οι OPOST, OLCUC (non-POSIX), ONLCR, OCRNL, ONOCR, ONLRET, OFILL, OFDEL (non-POSIX, non-LINUX), NLDLY, CRDLY, TABDLY, BSDLY, VTDLY, και FFDLY. Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990), η παράμετρος c_oflag, δεν χρησιμοποιείται. Ως εκ τούτου πρέπει να είναι πάντα **NUL** (0).

c_cflag : Η παράμετρος c_cflag ελέγχει τα τεχνικά χαρακτηριστικά της σειριακής θύρας RS-232 του Η/Υ, τα οποία πρέπει να συμφωνούν με τα τεχνικά χαρακτηριστικά της σειριακής θύρας της περιφερειακής συσκευής με την οποία χρειάζεται να αποκατασταθεί επικοινωνία . Υπάρχουν αρκετές δυνατές τιμές για την παράμετρο αυτή, οι οποίες ονομαστικά είναι οι CBAUD (non-POSIX), CBAUDEX (non-POSIX), CSIZE, CSTOPB, CREAD, PARENB, PARODD, HUPCL, CLOCAL, LOBLK (non-POSIX, non-LINUX), CIBAUD (non-POSIX), CRTSCTS (non-POSIX). Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990) χρησιμοποιούνται ως απαραίτητες οι τιμές CSIZE, CREAD, PARENB, HUPCL και CBAUD, ως εξής:

> **CSIZE:** Δίνεται η τιμή CS7, για μήκος χαρακτήρα 7 bits.
>
> **CREAD:** Ενεργοποίηση του δέκτη. Αν ο δέκτης δεν είναι ενεργοποιημένος, η θύρα RS-232 του Η/Υ μπορεί να αποστείλει χαρακτήρες αλλά δεν μπορεί να λάβει.

PARENB: Ενεργοποίηση του ελέγχου parity. Ένα parity-bit προστίθεται σε κάθε χαρακτήρα.

HUPCL: Η θύρας RS-232 θα αποσυνδεθεί όταν και η τελευταία process που χρησιμοποιεί την θύρα κλείσει το ειδικό αρχείο (με την κλήση **close**).

CBAUD: Δίνεται η τιμή B9600, για ταχύτητα μετάδοσης και λήψης δεδομένων 9600 baud. Ο δείκτης CBAUD δεν χρησιμοποιείται στα υποπρογράμματα, όπως μετατράπηκαν στα πλαίσια της παρούσης ΔΕ διότι ο δείκτης CBAUD δεν ακολουθεί το πρότυπο POSIX. Επιλέχθηκε άλλος τρόπος ρύθμισης της ταχύτητας μεταφοράς δεδομένων, ο οποίος θα παρουσιασθεί στη συνέχεια.

Επειδή για την παράμετρο c_cflag χρειάζεται να χρησιμοποιούνται περισσότεροι από ένας δείκτες, αυτοί χωρίζονται μεταξύ τους στο προγραμματιστικό επίπεδο με τον λογικό τελεστή Exclusive OR. (|).

c_lflag: Η παράμετρος c_lflag χρησιμοποιείται συνήθως όταν η επικοινωνία γίνεται κατά το canonical mode. Υπάρχουν αρκετές δυνατές τιμές για την παράμετρο αυτή, οι οποίες ονομαστικά είναι οι ISIG, ICANON, XCASE (non-POSIX, non-LINUX), ECHO, ECHOE, ECHOK, ECHONL, ECHOCTL (non-POSIX), ECHOPRT (non-POSIX), ECHOKE (non-POSIX), DEFECHO (non-POSIX, non-LINUX), FLUSHO (non-POSIX, non-LINUX), NOFLSH, TOSTOP, PENDIN (non-POSIX, non-LINUX), IEXTEN. Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990), η παράμετρος c_oflag, δεν χρησιμοποιείται. Ως εκ τούτου πρέπει να είναι πάντα **NUL** (0).

c_cc[NCCS]: 0 πίνακας c_cc μπορεί να περιέχει τους ακόλουθους ειδικούς χαρακτήρες, καθώς και κάποιους άλλους non-POSIX που δεν θα αναφερθούν:

	canonical mode	non-canonical mode
c_cc[VINTR]	**INTR**	**INTR**
c_cc[VQUIT]	**QUIT**	**QUIT**
c_cc[VERASE]	**ERASE**	
c_cc[VKILL]	**KILL**	
c_cc[VEOF]	**EOF**	

c_cc[VMIN]		**MIN**
c_cc[VEOL]	**EOL**	
c_cc[VTIME]		**TIME**
c_cc[VSTART]	**START**	**START**
c_cc[VSTOP]	**STOP**	**STOP**
c_cc[VSUSP]	**SUSP**	**SUSP**

όπου βέβαια, MIN και TIME δεν είναι ειδικοί χαρακτήρες αλλά τιμές για τον αριθμό χαρακτήρων που πρέπει να διαβασθούν σε καθορισμένο χρόνο αντίστοιχα, όπως σημειώθηκε και πιοπάνω.

Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990), χρησιμοποιούνται ως απαραίτητοι οι ειδικοί χαρακτήρες VMIN και VTIME, καθώς και οι VINTR, VQUIT, VERASE και VKILL. Παρόλαυτά οι ειδικοί χαρακτήρες VINTR και VQUIT δεν χρησιμοποιούνται στα υποπρογράμματα, όπως μετατράπηκαν στα πλαίσια της παρούσης ΔΕ διότι δεν χρειάζεται να χρησιμοποιηθεί ο δείκτης ISIG στην παράμετρο c_lflag. Επίσης, οι ειδικοί χαρακτήρες VERASE και VKILL δεν χρησιμοποιούνται στα υποπρογράμματα, διότι η επικοινωνία γίνεται κατά το non-canonical mode (βλ. και **man termios.h, man termios**)

cfsetospeed(): εντολή της **termios**, που ρυθμίζει την ταχύτητα μετάδοσης δεδομένων για την σειριακή θύρα RS-232 (βλ. και **man termios**). Η ταχύτητα μετάδοσης μπορεί να πάρει τις τιμές Bbaud, όπου baud = 0, 50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200 και 230400. Στην προγραμματιστική υλοποίηση της παρούσης ΔΕ Bbaud = B9600. Η χρήση της cfsetospeed() για την **termios.h** / POSIX αντικαθιστά το δείκτη CBAUD όπως υπήρχε για την **termio.h** / System V.

tcsetattr():εντολή της **termios**, που μεταφέρει στη σειριακή θύρα RS-232, τις ρυθμίσεις που γίνονται μέσω των παραμέτρων c_iflag, c_oflag, c_cflag, c_lflag, του πίνακα c_cc[NCCS], και της εντολής cfsetospeed(). Οι ρυθμίσεις αυτές γίνονται αν φυσικά το επιτρέπει το υλικό της σειριακής θύρας(βλ. και **man termios**). Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990) για την **termio.h** / System V, η υπόψιν μεταφορά μπορούσε να γίνει μόνο από την κλήση συστήματος **ioctl**. Στην προγραμματιστική υλοποίηση της παρούσας ΔΕ, αυτό μπορεί να γίνει τόσο με την

κλήση **ioctl**, όσο και με την εντολή **tcsetattr(),** που προτιμήθηκε για περισσότερη ευκολία.

tcgetattr():εντολή της **termios**, που μεταφέρει τις ήδη υπάρχουσες ρυθμίσεις της σειριακής θύρας RS-232, πίσω στον Η/Υ, για έλεγχο ή για χρήση (βλ. και **man termios**). Στην προγραμματιστική υλοποίηση της παρούσας ΔΕ, η tcgetattr() χρησιμοποιείται για την επαλήθευση των ρυθμίσεων που έχουν γίνει στη σειριακή θύρα RS-232, μέσω της tcsetattr()**.** Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990), τέτοιος έλεγχος δεν γίνεται.

tcflush():εντολή της **termios**, που αδειάζει τους καταχωρητές εισόδου και εξόδου της σειριακής θύρας RS-232 (βλ. και **man termios**). Στην προγραμματιστική υλοποίηση του Χαρβάτη (1990) για την **termio.h** / System V, η υπόψιν μεταφορά μπορούσε να γίνει μόνο από την κλήση συστήματος **ioctl**. Στην προγραμματιστική υλοποίηση της παρούσας ΔΕ, αυτό μπορεί να γίνει τόσο με την κλήση **ioctl**, όσο και με την εντολή tcflush()**,** που προτιμήθηκε για περισσότερη ευκολία.

4.4.3 Χρήση των συναρτήσεων unistd.h, fcntl.h και sys/time.h

- **unistd.h** (βλ. και **man unistd.h**)

  Συνάρτηση βιβλιοθήκης απαραίτητη για την ορθή λειτουργία της **termios.h** και των κλήσεων συστήματος **read**, **write**, **close**, **lockf**.

- **fcntl.h** (βλ. και **man fcntl.h, man fcntl**)

  Συνάρτηση βιβλιοθήκης απαραίτητη για την ορθή λειτουργία της κλήσης συστήματος **open**.

- **sys/time.h** (βλ. και **man sys_time.h**, **man gettimeofday**)

  Συνάρτηση βιβλιοθήκης απαραίτητη για την ορθή κλήσης της **gettimeofday()**.

4.5 Προγραμματιστική υλοποίηση από τον Δ. Χαρβάτη (1990)

4.5.1 Γενικά

Στη συνέχεια παρατίθενται ο κώδικας σε γλώσσα προγραμματισμού C, που αναπτύχθηκε και χρησιμοποιήθηκε από τον Χαρβάτη (1990) για περιβάλλον Λ/Σ UNIX / System V. Επειδή το κείμενο του κώδικα προέκυψε στην παρούσα ΔΕ ως αποτέλεσμα σάρωσης και αναγνώρισης χαρακτήρων, είναι δυνατόν να υπάρχουν λάθη, παρόλο που καταβλήθηκε προσπάθεια για τον περιορισμό τους. Επομένως, σε

περίπτωση αμφιβολίας, ο αναγνώστης πρέπει να αναφέρεται στο πρωτότυπο κείμενο της ΔΕ του Δ. Χαρβάτη (1990). Ο κώδικας αποτελείται από αρχεία **.c**, τα οποία περιέχουν υπορουτίνες αλλά από μόνα τους δεν συγκροτούν πλήρες πρόγραμμα. Για να χρησιμοποιηθούν οι υπορουτίνες, από ένα πρόγραμμα χρήστη θα πρέπει να συγκροτηθεί κατάλληλη βιβλιοθήκη. Τα σημεία του κώδικα που πρέπει να αλλάξουν, προκειμένου αυτός να προσαρμοσθεί για περιβάλλον Λ/Σ LINUX / POSIX. σημαίνονται με κατάλληλη σκίαση [ ].

4.5.2 Κώδικας

opendv.c

```
/* AUTHOR : D. Harvatis – March 88 */
#include <errno.h>
#include <unistd.h>
#include <fcntl.h>
#include <termio.h>
#include <stdio.h>
extern void SetCnt();/*SetCnt(dv,t) int dv,t; */
void ClearCnt(dv)
int dv;
{
      if ( ioctl(dv,TCFLSH,2) == -1) {
            fprintf(stderr,"ClearCnt : ioctl failed.\n");
                  exit(1);
      }
}
void OpenCnt(dv,df)
int *dv;
char *df;
{
      if ((*dv = open(df,O_RDWR)) < 0)  {
            if (errno==ENAMETOOLONG || errno==ENOTDIR
                  ||errno==ENOENT || errno==EISDIR )
                  fprintf(stderr,
                  "\nOpenCnt : bad file name.\n");
            else if (errno==EMFILE || errno==ENFILE)
                  fprintf(stderr,
                  "\nOpenCnt : too many open files.\n");
            else if (errno==EACCES || errno==EROFS ||errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : permission denied.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : open failed.\n");
            exit(1);
      }
      /* check if RS-232 is used by another program */
      if (lockf(*dv,F_TLOCK,0) !=0) {
            if (errno==EACCES || errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : line unavailable - try again later.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : unexpected lockf error.\n");
            exit(1);
      }
      SetCnt(*dv,-1);
      ClearCnt(*dv);
}
void CloseCnt(dv)
int dv;
{
      if (close(dv) !=0) {
            fprintf(stderr,"CloseCnt : close failed.\n");
            exit(1);
      }
}
```

modedv.c

```
/* AUTHOR : D. Harvatis – March 88 */

#include <termio.h>
#include <stdio.h>
void SetCnt(dv,t)
int dv,t;
{
      unsigned char c;
      struct termio arg;
      if (t != 0)
            c=1;
      else
            c=0;
      arg.c_iflag = IGNPAR;
      arg.c_oflag = 0;
      arg.c_cflag = B9600|CS7|CREAD|PARENB|HUPCL;
      arg.c_lflag = 0;
      arg.c_line = 0;
      arg.c_cc[VINTR] = CINTR;
      arg.c_cc[VQUIT] = CQUIT;
      arg.c_cc[VERASE] = CERASE;
      arg.c_cc[VKILL] = CKILL;
      arg.c_cc[VMIN] = c;
      arg.c_cc[VTIME] = 0;
      if ( ioctl(dv,TCSETA,&arg) == -1) {
            fprintf(stderr,"\nSetCnt : ioctl failed.\n");
            exit(1);
      }
}
```

chardv.c

```
/* AUTHOR : D. Harvatis – March 88 */

#include <termio.h>
#include <stdio.h>
void FromCnt(dv,c)
int dv;
char *c;
{
      int i;
      i= read(dv,c,1);
      if ( i== 0 ) *c = 0;
      else if ( i !=1) {
            fprintf(stderr,"Read error in FromCnt.\n");
            exit(1);
      }
}
void ToCnt(dv,c)
int dv;
char *c;
{
      if (write(dv,c,1) !=1) {
            fprintf(stderr,"Write error in ToCnt()\n");
            exit(1);
      }
}
```

datadv.c

```
/* AUTHOR : D. Harvatis – March 88 */

#include <stdio.h>
#define check(c) (((c)<060)||((c)>071))
extern void FromCnt(); /*FromCnt(dv,c) int dv; char *c; */
extern long atol();
void DataCnt(dv,err,n)
int dv;
int *err;
long *n;
{
      char s[6];
      int i;
      FromCnt(dv,s);
      if (s[0] == 041) *err = -1;
      else {
            *err=check(s[0]);
            for (i=1; i<6; ++i) {
                  FromCnt(dv,&s[i]);
                  *err += check(s[i]);
            }
            *n = atol(s);
      }
}
```

date.c

```
/* AUTHOR : D. Harvatis – March 88 */

#include <time.h>
#include <stdio.h>
void Time_to_I(s,m,h,D,M,Y)
int *s, *m, *h, *D, *M, *Y;
{
      long c;
      struct timeval now;
      struct timezone tzp;
      struct tm *Time;
      if ( gettimeofday(&now,&tzp) !=0) {
            fprintf(stderr,
            "\nTime_to_I : gettimeofday system call failed.\n");
            exit(1);
      } else {
            c = (long) now.tv_sec;
            Time = localtime(&c);
            *s = (*Time).tm_sec;
            *m = (*Time).tm_min;
            *h = (*Time).tm_hour;
            *D = (*Time).tm_mday;
            *M = (*Time).tm_mon + 1;
            *Y = (*Time).tm_year;
            }
}
```

secs.c

```
/* AUTHOR : D. Harvatis – March 88 */

#include <time.h>
#include <stdio.h>
void Ellapsed(sref,tref,s,t)
long *sref, *tref, *s, *t;
{
      struct timeval now;
      struct timeval zone;
      if (gettimeofday(&now,&zone) !=0) {
            fprintf(stderr,
            "\nEllapsed : gettimeofaday system call failed.\n");
            exit(1);
      } else {
            if (now.tv_usec < *tref) {
                  now.tv_usec += 1000000;
                  now.tv_sec--;
            }
            *t = now.tv_usec - *tref;
            *s = now.tv_sec - *sref;
      }
}
```

4.6 Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας

4.6.1 Γενικά

Στη συνέχεια παρατίθενται ο κώδικας σε γλώσσα προγραμματισμού C, που βασίζεται στον αντίστοιχο του Χαρβάτη (1990). Ο κώδικας αποτελείται από αρχεία **.c**, όπως στην προηγούμενη ενότητα, τα οποία περιέχουν υπορουτίνες αλλά από μόνα τους δεν συγκροτούν πλήρες πρόγραμμα. Για να χρησιμοποιηθούν οι υπορουτίνες, από ένα πρόγραμμα χρήστη θα πρέπει να συγκροτηθεί κατάλληλη βιβλιοθήκη. Πραγματοποιήθηκαν αλλαγές για την προσαρμογή του σε περιβάλλον Λ/Σ LINUX / POSIX. Οι διαφορές που προέκυψαν στα προγράμματα, μετά τη μετατροπή, δηλαδή οι προσθέσεις, αφαιρέσεις και αλλαγές σημαίνονται με κατάλληλη σκίαση [ ]. Τα αρχεία του υπόψιν κώδικα περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία.

4.6.2 Κώδικας

opendv.c

```
/* AUTHOR : D. Harvatis – March 88 */
/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <errno.h>
#include <unistd.h>
#include <fcntl.h>
#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
extern void SetCnt();/*SetCnt(dv,t) int dv,t; */
struct termios oldset;
void ClearCnt(dv)
int dv;
{
      if ( tcflush(dv,TCIOFLUSH) == -1) {
            fprintf(stderr,"ClearCnt : tcflush failed.\n");
            exit(1);
      }
}
void OpenCnt(dv,df)
int *dv;
char *df;
{
      if ((*dv = open(df,O_RDWR)) < 0) {
      if (errno==ENAMETOOLONG || errno==ENOTDIR
            ||errno==ENOENT || errno==EISDIR )
            fprintf(stderr,
            "\nOpenCnt : bad file name.\n");
      else if (errno==EMFILE || errno==ENFILE)
            fprintf(stderr,
            "\nOpenCnt : too many open files.\n");
      else if (errno==EACCES || errno==EROFS ||errno==EAGAIN)
            fprintf(stderr,
            "\nOpenCnt : permission denide.\n");
      else
            fprintf(stderr,
            "\nOpenCnt : open failed.\n");
      exit(1);
      }
      /* check if RS-232 is used by another program */
      if (lockf(*dv,F_TLOCK,0) !=0) {
            if (errno==EACCES || errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : line unavailable - try again later.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : unexpected lockf error.\n");
            exit(1);
      }
      {tcgetattr(*dv,&oldset);
      }
SetCnt(*dv,-1);
ClearCnt(*dv);
}
void CloseCnt(dv)
int dv;
{tcsetattr(dv,TCSANOW,&oldset);
```

```
{
	if (close(dv) !=0) {
		fprintf(stderr,"CloseCnt : close failed.\n");
		exit(1);
	}
}
}
```

modedv.c

```
/* AUTHOR : D. Harvatis – March 88 */
/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <errno.h>
#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
void SetCnt(dv,t)
int dv,t;
{
	unsigned char c;
	struct termios arg, check;
	if (t != 0)
		c=1;
	else
		c=0;
	arg.c_iflag = IGNPAR;
	arg.c_oflag = 0;
	arg.c_cflag = CS7|CREAD|PARENB|HUPCL;
	arg.c_lflag = 0;
	arg.c_cc[VMIN] = c;
	arg.c_cc[VTIME] = 0;
	if(cfsetospeed(&arg,B9600)!=0){
		fprintf(stderr,"\nSetCnt: failed to set port speed.\n");
			exit(1);
	}
	if(tcsetattr(dv,TCSANOW,&arg)==-1){
		fprintf(stderr,"\nSetCnt : tcsetattr failed.\n");
			exit(1);
	}
	tcgetattr(dv,&check);
	if(
		check.c_iflag != arg.c_iflag ||
		check.c_oflag != arg.c_oflag ||
		check.c_lflag != arg.c_lflag ||
		check.c_cc[VMIN] != arg.c_cc[VMIN] ||
		check.c_cc[VTIME] != arg.c_cc[VTIME] ||
		cfgetospeed(&check) != cfgetospeed(&arg)
	){
	fprintf(stderr,"\nSetCnt: failed to set port parameters.\n");
	exit(1);
	}
}
```

chardv.c

```
/* AUTHOR : D. Harvatis – March 88 */
/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
void FromCnt(dv,c)
int dv;
char *c;
{
      int i;
      i= read(dv,c,1);
      if ( i== 0 ) *c = 0;
      else if ( i !=1) {
            fprintf(stderr,"Read error in FromCnt.\n");
            exit(1);
      }
}
void ToCnt(dv,c)
int dv;
char *c;
{
      if (write(dv,c,1) !=1) {
            fprintf(stderr,"Write error in ToCnt()\n");
            exit(1);
      }
}
```

datadv.c

```
/* AUTHOR : D. Harvatis – March 88 */
/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <stdio.h>
#include <stdlib.h>
#define check(c) (((c)<060)||((c)>071))
extern void FromCnt(); /*FromCnt(dv,c) int dv; char *c; */
void DataCnt(dv,err,n)
int dv;
int *err;
long *n;
{
      char s[6];
      int i;
      FromCnt(dv,s);
      if (s[0] == 041) *err = -1;
      else {
            *err=check(s[0]);
            for (i=1; i<6; ++i) {
                  FromCnt(dv,&s[i]);
                  *err += check(s[i]);
            }
            *n = atol(s);
      }
}
```

date.c

```
/* AUTHOR : D. Harvatis – March 88 */
/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <stdio.h>
#include <time.h>
#include <sys/time.h>
#include <stdlib.h>
void Time_to_I(s,m,h,D,M,Y)
int *s, *m, *h, *D, *M, *Y;
{
      long c;
      struct timeval now;
      struct timezone tzp;
      struct tm *Time;
      if ( gettimeofday(&now,&tzp) !=0) {
            fprintf(stderr,
            "\ntime_to I : gettimeofday system call failed.\n");
            exit(1);
      } else {
            c = (long) now.tv_sec;
            Time = localtime(&c);
            *s = (*Time).tm_sec;
            *m = (*Time).tm_min;
            *h = (*Time).tm_hour;
            *D = (*Time).tm_mday;
            *M = (*Time).tm_mon + 1;
            *Y = (*Time).tm_year;
            }
}
```

secs.c

```
/* AUTHOR : D. Harvatis – March 88 */

/* MODIFIED : G.D. Alexopoulos – April 08 */
#include <stdio.h>
#include <time.h>
#include <sys/time.h>
#include <stdlib.h>
void Ellapsed(sref,tref,s,t)
long *sref, *tref, *s, *t;
{
      struct timeval now;
      struct timeval zone;
      if (gettimeofday(&now,&zone) !=0) {
            fprintf(stderr,
            "\nEllapsed : gettimeofaday system call failed.\n");
            exit(1);
      } else {
            if (now.tv_usec < *tref) {
                  now.tv_usec += 1000000;
                  now.tv_sec--;
            }
            *t = now.tv_usec - *tref;
            *s = now.tv_sec - *sref;
      }
}
```

4.7 Ανάλυση κώδικα

4.7.1 Εισαγωγή

Ο κώδικας σύμφωνα με τη προγραμματιστική υλοποίηση που πρότεινε ο Χαρβάτης (1990), η οποία και στην ουσία της ακολουθείται και για την προγραμματιστική υλοποίηση της παρούσας ΔΕ, αποτελείται, όπως φάνηκε και στα προηγούμενα, από έξι (6) αρχεία με τμήματα προγράμματος σε γλώσσα προγραμματισμού C, τα οποία περιέχουν εννέα (9) υπορουτίνες. Οι υπορουτίνες αυτές χωρίζονται σε τρεις κατηγορίες (α) τις υπορουτίνες ρύθμισης, (β) τις υπορουτίνες επικοινωνίας και (γ) τις γενικές υπορουτίνες, ως εξής:

Υπορουτίνες ρύθμισης

Οι υπορουτίνες αυτές χρησιμοποιούνται για τον καθορισμό των λειτουργικών παραμέτρων της θύρας RS-232 και περιέχονται στα προγράμματα **opendv.c** και **modedv.c**:

- Αρχείο - Πρόγραμμα **opendv.c**

  Περιέχει τις υπορουτίνες **ClearCnt, OpenCnt** και **CloseCnt**.

- Αρχείο - Πρόγραμμα **mode.dv**

  Περιέχει την υπορουτίνα **SetCnt.**

Υπορουτίνες επικοινωνίας

Οι υπορουτίνες αυτές φροντίζουν για την χαμηλού επιπέδου επικοινωνία με τη συνδεδεμένη σειριακή συσκευή και περιέχονται στα προγράμματα **chardv.c** και **datadv.c**.

- Αρχείο - Πρόγραμμα **chardv.c**

  Περιέχει τις υπορουτίνες **FromCnt** και **ToCnt.**

- Αρχείο - Πρόγραμμα **datadv.c**

  Περιέχει την υπορουτίνα **DataCnt.**

Γενικές υπορουτίνες

Οι υπορουτίνες αυτές δεν έχουν σχέση με τη συνδεδεμένη σειριακή συσκευή και εκτελούν ορισμένες γενικές λειτουργίες όπως π.χ. ανάγνωση από το ρολόι του Η/Υ της ημερομηνίας και της ώρας και περιέχονται στα προγράμματα **date.c** και **secs.c**.

- Αρχείο - Πρόγραμμα **date.c**

  Περιέχει την υπορουτίνα **Time_to_I.**

- Αρχείο - Πρόγραμμα **secs.c**

  Περιέχει την υπορουτίνα **Ellapsed.**

Στη συνέχεια θα περιγραφούν οι υπόψιν υπορουτίνες.

4.7.2 OpenCnt

Σύνταξη: **void OpenCnt (dv, df)**

int *dv;

char *df;

Περιγραφή: Είναι υπορουτίνα ρύθμισης. Η υπορουτίνα **OpenCnt** ανοίγει το ειδικό αρχείο της θύρας RS-232 (στην εδώ περίπτωση το /dev/ttyS0) ως περιφερειακό του Η/Υ, ελέγχει αν η θύρα αυτή είναι ελεύθερη, ρυθμίζει τα χαρακτηριστικά της θύρας όπως απαιτείται για την επικοινωνία με κάποια σειριακή συσκευή, της οποίας τα χαρακτηριστικά είναι ήδη ορισμένα (με την κλήση της υπορουτίνας **SetCnt**) και αδειάζει τους καταχωρητές εισόδου και εξόδου (με την κλήση της υπορουτίνας **ClearCnt**). Η **OpenCnt** είναι η υπορουτίνα που αποκαθιστά την σειριακή σύνδεση και συνεπώς είναι απαραίτητο να καλείται στην αρχή ενός προγράμματος, πριν από οποιαδήποτε άλλη υπορουτίνα επικοινωνίας.

Μεταβλητές Εισόδου: **char *df;**

df είναι η διεύθυνση ενός **Null-terminated string** χαρακτήρων που περιέχει το όνομα του ειδικού αρχείου που αντιστοιχεί στην θύρα RS-232 στην οποία έχει συνδεθεί μία σειριακή συσκευή, με την οποία χρειάζεται να αποκατασταθεί επικοινωνία.

Μεταβλητές Εξόδου: **int *dv;**

dv είναι η διεύθυνση που όταν επιστρέψει η **OpenCnt**, θα περιέχει το file descriptor. Υπενθυμίζεται ότι το file descriptor είναι ο αριθμός που επιστρέφει η κλήση open όταν ανοίγει ένα αρχείο και είναι απαραίτητος για κάθε περαιτέρω επικοινωνία με το αρχείο αυτό. Έτσι, ο αριθμός αυτός χρειάζεται σε όλες τις άλλες υπορουτίνες ρύθμισης και επικοινωνίας και αποτελεί γι' αυτές παράμετρο εισόδου.

Σχόλια: Στο εσωτερικό της **OpenCnt**, για το άνοιγμα του ειδικού αρχείου, χρησιμοποιείται η κλήση **open**. Αν η κλήση αυτή αποτύχει, εμφανίζεται στην οθόνη ένα μήνυμα και το πρόγραμμα τερματίζεται. Πιθανές αιτίες αποτυχίας της **open** και αντίστοιχα μηνύματα είναι:

- Να υπάρχει κάποιο λάθος στο όνομα του ειδικού αρχείου. Τυπώνεται το μήνυμα:

OpenCnt : bad file name.

- Ο αριθμός των ήδη ανοιχτών αρχείων να έχει φτάσει στο ανώτατο όριο που τίθεται από το σύστημα. Τυπώνεται το μήνυμα:

OpenCnt : too many open files.

- Ο χρήστης να μην έχει άδεια να ανοίξει το αρχείο αυτό. Τυπώνεται το μήνυμα:

OpenCnt : permission denied.

- Αν δεν συμβαίνει τίποτα από τα παραπάνω, αλλά παρόλαυτά υπάρχει αποτυχία της κλήσης **open**, εμφανίζεται το μήνυμα :

OpenCnt : open failed.

Στη συνέχεια, με την κλήση **lockf**, ελέγχεται αν η θύρα RS-232 είναι ελεύθερη και αν πράγματι είναι, κλειδώνεται προς χρήση. Αν υπάρχει κάποια άλλη process που την χρησιμοποιεί και την έχει κλειδώσει νωρίτερα, το πρόγραμμα τερματίζεται αφού εμφανισθεί στην οθόνη το μήνυμα :

OpenCnt : line anavailable - try again later.

Αν η **lockf** αποτύχει για κάποιο άλλο λόγο, εμφανίζεται στην οθόνη το μήνυμα :

OpenCnt : unexpected lockf error.

και το πρόγραμμα τερματίζεται.

Τέλος, καλούνται δύο άλλες υπορουτίνες, η **SetCnt** και η **ClearCnt**, οι οποίες ρυθμίζουν την RS-232 και αδειάζουν τους καταχωρητές εισόδου και εξόδου αντίστοιχα. Οι υπορουτίνες αυτές θα περιγραφούν με λεπτομέρεια στη συνέχεια.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **OpenCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **termios.h** αντί των επικεφαλίδων **termio.h** (για λόγους που έχουν ήδη εξηγηθεί)
- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)
- Χρήση της δομής **termios** [με το συνδυασμό των εντολών struct termios oldset; και tcgetattr(*dv,&oldset); αποθηκεύονται προς μελλοντική χρήση (δηλαδή ένα "reset" μέσα από τον κώδικα), οι ρυθμίσεις της θύρας RS-232, που υπήρχαν στον Η/Υ, πριν τις ρυθμίσεις που θα επιβληθούν από τις σχετικές υπορουτίνες του κώδικα]

4.7.3 SetCnt

Σύνταξη: **void SetCnt (dv, t)**

int dv, t;

Περιγραφή: Είναι υπορουτίνα ρύθμισης. Η υπορουτίνα αυτή ρυθμίζει τα λειτουργικά χαρακτηριστικά της θύρας RS-232 του Η/Υ, σύμφωνα με τις προδιαγραφές της σειριακής συσκευής που θα συνδεθεί, ώστε να αποκατασταθεί επικοινωνία. Στον κώδικα που παρουσιάζεται εμπεριέχονται οι επόμενες ρυθμίσεις (οι οποίες αφορούν π.χ. σε τυπικές προδιαγραφές της μονάδας επικοινωνίας CANBERRA – Model 1788):

Ταχύτητα Μετάδοσης 9600 baud.

Μήκος χαρακτήρα 7 bits.

Parity EVEN.

Ένα stop bit.

Στην **SetCnt**, η σειριακή επικοινωνία ρυθμίζεται κατά τον μη κανονικό τρόπο. Ο διακόπτης **TIME** τίθεται πάντα **0** ενώ ο διακόπτης **MIN** μπορεί να πάρει την τιμή **1** (οπότε η κλήση **read** δεν επιστρέφει αν δεν υπάρχει ένας τουλάχιστον χαρακτήρας προς ανάγνωση) ή **0** (οπότε η κλήση **read** επιστρέφει αμέσως).

Μεταβλητές Εισόδου: **int dv;**

int t;

dv είναι ένας ακέραιος που περιέχει το file descriptor. Τονίζεται ότι δεν είναι η διεύθυνση αλλά απ' ευθείας η τιμή του.

Η παράμετρος **t** είναι επίσης ακέραιος. Αν είναι ίση με μηδέν (0) τότε ο διακόπτης **MIN** τίθεται **0** (άμεση επιστροφή της κλήσης **read**) αλλιώς τίθεται **1** (η **read** περιμένει).

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **SetCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, δεν χρησιμοποιεί την κλήση **ioctl** και χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **errno.h** (προκειμένου να λειτουργούν οι έλεγχοι του αν έχουν μεταβιβασθεί οι επιθυμητές ρυθμίσεις στη θύρα)
- Χρήση των επικεφαλίδων **termios.h** αντί των επικεφαλίδων **termio.h** (για λόγους που έχουν ήδη εξηγηθεί)
- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)
- Χρήση της δομής **termios** αντί της δομής **termio** (για λόγους που έχουν ήδη εξηγηθεί)
- Η παράμετρος c_line, δεν υπάρχει για την δομή termios και επομένως δεν χρησιμοποιείται
- Η παράμετρος c_cflag, δεν μπορεί να ρυθμίσει την ταχύτητα της σειριακής θύρας (baud rate) στα πλαίσια της δομής **termios**, όπως γινόταν στα πλαίσια της δομής **termio**. Για τη ρύθμιση της ταχύτητας χρησιμοποιείται, όπως έχει σημειωθεί η εντολή cfsetospeed(). Αν η εντολή αυτή αποτύχει, το μήνυμα

  SetCnt : failed to set port speed.

  εμφανίζεται στην οθόνη και το πρόγραμμα τερματίζεται.

- Για την μετάδοση των ρυθμίσεων που έχουν ορισθεί για τη δομή termios στην σειριακή θύρα RS-232, χρησιμοποιείται η εντολή tcsetattr(). Αν η εντολή αυτή αποτύχει το μήνυμα

  SetCnt : tcsetattr failed.

  εμφανίζεται στην οθόνη και το πρόγραμμα τερματίζεται.

- Για τον έλεγχο του αν πράγματι ισχύουν οι ρυθμίσεις που μεταδόθηκαν στην RS-232 με την προηγούμενη εντολή tcsetattr(), χρησιμοποιείται η εντολή tcgetatrr() και αντίστοιχοι έλεγχοι με if. Αν ο έλεγχος αποτύχει το μήνυμα

  SetCnt : failed to set port parameters.

  εμφανίζεται στην οθόνη και το πρόγραμμα τερματίζεται.

Σημείωση: Όταν η υπορουτίνα **SetCnt** καλείται από την **OpenCnt**, η μεταβλητή t είναι ≠0 (στην εδώ υλοποίηση –1), δηλαδή ο δείκτης MIN ρυθμίζεται έτσι ώστε η κλήση **read** να περιμένει έως ότου υπάρξει στον καταχωρητή εισόδου ένας τουλάχιστον χαρακτήρας.

4.7.4 C1earCnt

Σύνταξη: **void ClearCnt (dv)**

int dv;

Περιγραφή: Είναι υπορουτίνα ρύθμισης. Η υπορουτίνα **ClearCnt** αδειάζει τους καταχωρητές εισόδου και εξόδου της θύρας RS-232.

Μεταβλητές Εισόδου: **int dv;**

dv είναι ένας ακέραιος που περιέχει το file descriptor.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **ClearCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, δεν χρησιμοποιεί την κλήση **ioctl** και χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **termios.h** αντί των επικεφαλίδων **termio.h** (για λόγους που έχουν ήδη εξηγηθεί)

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)

- Χρήση της δομής **termios** αντί της δομής **termio** (για λόγους που έχουν ήδη εξηγηθεί)

- Για το άδειασμα των ουρών εισόδου και εξόδου της θύρας RS-232, χρησιμοποιείται η εντολή tcflush(). Αν η εντολή αυτή αποτύχει το μήνυμα

ClearCnt : tcflush failed.

εμφανίζεται στην οθόνη και το πρόγραμμα τερματίζεται.

4.7.5 C1oseCnt

Σύνταξη: **void CloseCnt (dv)**

int dv;

Περιγραφή: Είναι υπορουτίνα ρύθμισης. Η **CloseCnt** κλείνει το ειδικό αρχείο της θύρας RS-232 τερματίζοντας έτσι τη σύνδεση. Η υπορουτίνα αυτή θα πρέπει να καλείται μόνο αφού έχει ολοκληρωθεί η επικοινωνία με τη σειριακή συσκευή.

Μεταβλητές Εισόδου: **int dv;**

dv είναι ένας ακέραιος που περιέχει το file descriptor.

Σχόλια: Το κλείσιμο του ειδικού αρχείου γίνεται με την κλήση συστήματος **close**. Αν η **close** αποτύχει, εμφανίζεται το μήνυμα

CloseCnt : close failed.

και το πρόγραμμα τερματίζεται.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **CloseCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **termios.h** αντί των επικεφαλίδων **termio.h** (για λόγους που έχουν ήδη εξηγηθεί)

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)

- Χρήση της δομής **termios** [με το συνδυασμό των εντολών struct termios oldset; και tcsetattr(*dv,TCSANOW,&oldset); αποστέλλονται στη θύρα RS-232 οι ρυθμίσεις της, που υπήρχαν στον Η/Υ, πριν τις ρυθμίσεις που επιβλήθηκαν από

τις σχετικές υπορουτίνες του κώδικα, αρκεί αυτές να είχαν αποθηκευθεί από πριν με την εντολή tcgetattr(*dv,&oldset);]]

4.7.6 ToCnt

Σύνταξη: **void ToCnt (dv, c)**

int dv;

char *c;

Περιγραφή: Είναι υπορουτίνα επικοινωνίας. Η υπορουτίνα **ToCnt** αποστέλλει, μέσω της θύρας RS-232, ένα χαρακτήρα στη σειριακή συσκευή που είναι συνδεδεμένη με τη θύρα.

Μεταβλητές Εισόδου: **int dv;**

char *c;

dv είναι ένας ακέραιος που περιέχει το file descriptor.

c είναι η διεύθυνση του προς αποστολή χαρακτήρα.

Σχόλια: Η υπορουτίνα **ToCnt** χρησιμοποιεί την κλήση **write** για να στείλει τον χαρακτήρα στην RS-232. Αν η **write** αποτύχει, εμφανίζεται το μήνυμα

Write error in ToCnt.

και το πρόγραμμα τερματίζεται.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **ToCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται την εξής αλλαγή:

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)

4.7.7 FromCnt

Σύνταξη: **void FromCnt (dv, c)**

int dv;

char *c;

Περιγραφή: Είναι υπορουτίνα επικοινωνίας. Η υπορουτίνα **FromCnt** διαβάζει ένα χαρακτήρα από τον καταχωρητή εισόδου της θύρας RS-232 (τον χαρακτήρα αυτόν υποτίθεται ότι τον έχει αποστείλει η σειριακή συσκευή που είναι συνδεδεμένη με τη

θύρα). Ανάλογα με το πώς έχει ρυθμιστεί η θύρα, η **FromCnt** είτε περιμένει έως ότου υπάρξει διαθέσιμος χαρακτήρας, είτε επιστρέφει αμέσως άσχετα με το αν υπάρχει χαρακτήρας ή όχι.

Μεταβλητές Εισόδου: **int dv;**

dv είναι ένας ακέραιος που περιέχει το file descriptor.

Μεταβλητές Εξόδου: **char *c;**

c είναι η διεύθυνση όπου τοποθετείται ο χαρακτήρας που φθάνει από τη συνδεδεμένη σειριακή συσκευή. Στην περίπτωση που η θύρα είναι ρυθμισμένη για άμεση επιστροφή της **read** και ο καταχωρητής εισόδου είναι άδειος, στην θέση **c** τοποθετείται ο χαρακτήρας **NUL** (0).

Σχόλια: Για το διάβασμα του χαρακτήρα χρησιμοποιείται η κλήση **read**. Όπως και στις άλλες περιπτώσεις, αν η **read** αποτύχει, εμφανίζεται το μήνυμα

Read error in FromCnt.

και το πρόγραμμα τερματίζεται.

Επιστροφή από την **FromCnt** χαρακτήρα **NUL**, σημαίνει ότι δεν υπήρχε διαθέσιμος χαρακτήρας από τη συνδεδεμένη σειριακή συσκευή (αυτό μπορεί να συμβεί μόνο αν η θύρα είναι ρυθμισμένος για άμεση επιστροφή της **read**). Η εκλογή του χαρακτήρα **NUL** γι' αυτό το σκοπό είναι ασφαλής όταν το **NUL** δεν συγκαταλέγεται ανάμεσα στους χαρακτήρες που μπορεί να στείλει η συνδεδεμένη σειριακή συσκευή.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **ToCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται την εξής αλλαγή:

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)

4.7.8 DataCnt

Σύνταξη: **void DataCnt (dv, err, n)**

int dv;

int *err;

long *n;

Περιγραφή: Είναι υπορουτίνα επικοινωνίας. Η υπορουτίνα **DataCnt** με διαδοχικές κλήσεις της **FromCnt** διαβάζει έως 6 χαρακτήρες, τους οποίους μπορεί να έχει στείλει η σειριακή συσκευή στη θύρα RS-232[4]. και τους μετατρέπει σε έναν "μακρύ" (στα αγγλικά long) ακέραιο.

Μεταβλητές Εισόδου: **int dv;**

dv είναι ένας ακέραιος που περιέχει το file descriptor.

Μεταβλητές Εξόδου: **int *err;**

long *n;

err είναι η διεύθυνση ενός ακεραίου όπου τοποθετείται ένας κωδικός που δείχνει αν η **DataCnt** πέτυχε ή όχι[5]. Αν ένας ή περισσότεροι από τους έξι χαρακτήρες που αποτελούν τη μέτρηση δεν είναι δεκαδικά ψηφία (0 έως 9), ο κωδικός αυτός είναι ένας θετικός αριθμός ίσος με το πλήθος των χαρακτήρων που παρουσίασαν λάθος.

n είναι η διεύθυνση ενός "μακρύ" ακεραίου (4 bytes) όπου θα τοποθετηθεί το αποτέλεσμα της μέτρησης.

Σχόλια: Για το διάβασμα των έξι χαρακτήρων η **DataCnt** καλεί την υπορουτίνα **FromCnt**. Στην συνέχεια, οι χαρακτήρες αυτοί ελέγχονται για το αν είναι δεκαδικά ψηφία ή όχι με το macro check. Τέλος, οι έξι χαρακτήρες μετατρέπονται σε ακέραιο με την βοήθεια της συνάρτησης **atol** (βλ και **man atol**, **man stdlib.h**).

Η **DataCnt** δεν ελέγχει αν ο χαρακτήρας που επέστρεψε η **FromCnt** είναι **NUL**, δηλαδή την περίπτωση να μην υπάρχει άμεσα διαθέσιμος χαρακτήρας από τη συνδεδεμένη σειριακή συσκευή. Για το λόγο αυτό, πρέπει, όταν χρησιμοποιείται η **DataCnt**, να έχει ρυθμισθεί η θύρα RS-232 έτσι ώστε η κλήση **read** να μην επιστρέφει άμεσα αλλά να περιμένει έως ότου υπάρξει χαρακτήρας στον καταχωρητή εισόδου.

[4]Αν η σειριακή συσκευή είναι π.χ. η μονάδα επικοινωνίας CANBERRA – Model 1788, οι 6 χαρακτήρες προς ανάνωση μπορεί να είναι ψηφία (δηλαδή χαρακτήρες ASCII 0, 1, 2, 3, 4, 5, 6, 7, 8, 9) (βλ. και το επόμενο Κεφάλαιο 5).

[5] Αν η σειριακή συσκευή είναι π.χ. η μονάδα επικοινωνίας CANBERRA – Model 1788, τότε σε περίπτωση επιτυχίας ο κωδικός αυτός είναι 0. Αν ο πρώτος χαρακτήρας που διαβάζει η **DataCnt** είναι ο χαρακτήρας **DONE** (!), πράγμα που σημαίνει ότι δεν υπάρχει άλλη μετρητική διάταξη να στείλει μέτρηση, ο κωδικός αυτός είναι –1 (βλ. και το επόμενο Κεφάλαιο 5).

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **DataCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται την εξής αλλαγή:

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της συνάρτησης atol)

4.7.9 Time_to_I

Σύνταξη: **void Time_to_I (s, m, h, D, M, Y)**

int *s, *m, *h, *D, *M, *Y;

Περιγραφή: Είναι γενική υπορουτίνα. Η υπορουτίνα **Time_to_I** χρησιμοποιείται για την ανάγνωση της ώρας και της ημερομηνίας από το ρολόι του Η/Υ.

Μεταβλητές Εξόδου: **int *s, *m, *h, *D, *M, *Y;**

s, **m** και **h** είναι οι διευθύνσεις όπου τοποθετούνται τα δευτερόλεπτα, τα λεπτά και οι ώρες αντίστοιχα. **D**, **M** και **Y** είναι οι διευθύνσεις όπου τοποθετούνται η μέρα, ο μήνας και ο χρόνος αντίστοιχα.

Σχόλια: Για την ανάγνωση της ημερομηνίας και της ώρας, η **Time_to_I** χρησιμοποιεί την εντολή gettimeofday(). Αν αυτή αποτύχει, εμφανίζεται το μήνυμα

Time_to_I : gettimeofday system call failed.

και το πρόγραμμα τερματίζεται.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **ToCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)
- Χρήση των επικεφαλίδων **sys/time.h** [για τη σωστή λειτουργία της εντολής gettimeofday()]

4.7.10 Ellapsed

Σύνταξη: **void Ellapsed (sref, tref, s, t)**

long *sref, *tref, *s, *t;

Περιγραφή: Είναι γενική υπορουτίνα. Η υπορουτίνα αυτή χρησιμοποιείται για τη μέτρηση του χρονικού διαστήματος που έχει περάσει από κάποια στιγμή αναφοράς.

Μεταβλητές Εισόδου: **long *sref, *tref;**

sref και **tref** είναι οι διευθύνσεις δύο "μακρών" ακεραίων. Οι τιμές που περιέχονται στις διευθύνσεις αυτές ορίζουν την χρονική στιγμή αναφοράς. Η **sref** περιέχει δευτερόλεπτα, ενώ η **tref** εκατομμυριοστά του δευτερολέπτου.

Μεταβλητές Εξόδου: **long *s, *t;**

s και **t** είναι διευθύνσεις δύο "μακρών" ακεραίων. Στις διευθύνσεις αυτές τοποθετούνται αντίστοιχα τα δευτερόλεπτα και τα εκατομμυριοστά δευτερολέπτου που μεσολάβησαν από τη στιγμή αναφοράς ως τη στιγμή κλήσης της **Ellapsed**.

Σχόλια: Για να μετρηθεί το χρονικό διάστημα που μεσολαβεί μεταξύ δύο χρονικών σημείων Α και Β ενός προγράμματος η **Ellapsed** καλείται δύο φορές :

1. Πρώτη κλήση στο χρονικό σημείο Α, δίνοντας στιγμή αναφοράς 0 sec, 0 μsec.

2. Δεύτερη κλήση στο χρονικό σημείο Β, δίνοντας στιγμή αναφοράς τις τιμές των sec και μsec που επέστρεψε η **Ellapsed** κατά την πρώτη κλήση. Τα sec και μsec που επιστρέφει τώρα, είναι το διάστημα που έχει μεσολαβήσει από το σημείο Α.

Η λειτουργία της **Ellapsed** βασίζεται στην εντολή gettimeofday().Αν αυτή αποτύχει, εμφανίζεται το μήνυμα

Ellapsed : gettimeofday system call failed.

και το πρόγραμμα τερματίζεται.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **ToCnt**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

- Χρήση των επικεφαλίδων **stdlib.h** (για τη σωστή λειτουργία της εντολής exit)
- Χρήση των επικεφαλίδων **sys/time.h** [για τη σωστή λειτουργία της εντολής gettimeofday()]

4.8 Χρήση κώδικα

4.8.1 Γενικά

Στην ενότητα αυτή, παρουσιάζονται τα αναγκαία για τη χρήση του κώδικα. Ειδικότερα, αντιμετωπίζονται θέματα όπως κλήση των υπορουτινών σε γλώσσα προγραμματισμού C, από προγράμματα σε γλώσσα FORTRAN, παρουσιάζονται οι αναγκαίες εντολές μετάφρασης προγραμμάτων σε C και FORTRAN, με έμφαση σε ειδικούς διακόπτες του μεταφραστή FORTRAN και τέλος εξετάζεται ο τρόπος συγκρότησης βιβλιοθήκης που να περιέχει τις υπορουτίνες FORTRAN και C, ώστε ο χρήστης, να χρειάζεται απλώς να συνδέεται μόνο με τη βιβλιοθήκη και όχι με τις αναγκαίες για αυτόν υπορουτίνες μία προς μία.

4.8.2 Κλήση από FORTRAN

Για να μπορεί ένα πρόγραμμα γραμμένο σε FORTRAN να καλέσει μία υπορουτίνα γραμμένη σε γλώσσα C, η οποία έχει άλλο όνομα στο καλούν πρόγραμμα και άλλο στην πραγματικότητα, πρέπει να του δοθούν μέσω της εντολής **#define** (η οποία στο Λ/Σ LINUX είναι ισοδύναμη με την **$ALIAS** του Λ/Σ HP-UX), οι παρακάτω πληροφορίες:

1. Το όνομα με το οποίο θα είναι γνωστή η υπορουτίνα στο πρόγραμμα FORTRAN.
2. Το όνομα με το οποίο θα είναι γνωστή η υπορουτίνα στο πρόγραμμα C.
3. Ο αριθμός των μεταβλητών της υπορουτίνας.
4. Ο τρόπος με τον οποίο θα περνά κάθε μεταβλητή, δηλαδή, αν θα μεταφέρεται από και προς την υπορουτίνα η διεύθυνση της παραμέτρου (by reference) ή η τιμή της (by value).

Το πρόγραμμα FORTRAN που καλεί υπορουτίνες C, ή άλλες, μέσω της εντολής "προεπεξεργασίας" (στα αγγλικά preprocessing) **#define**, θα πρέπει να μεταφράζεται από την εντολή **g77** με τη χρήση του διακόπτη

-x f77-cpp-input

Ισοδύναμα, αν το αρχείο που περιέχει το υπόψιν πρόγραμμα FORTRAN, έχει extension **".F"**, η χρήση του διακόπτη -x f77-cpp-input, δεν είναι απαραίτητη (βλ. και **man g77**)

Κλήση της **OpenCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **OpenCnt**, π.χ. με το FORTRAN όνομα OPENCNT, τότε μέσα στο πρόγραμμα πρέπει να υπάρχει η εντολή:

CALL OPENCNT(IDV, TD)

όπου, η μεταβλητή **TD**, στη εδώ περίπτωση μπορεί να είναι το αρχείο της θύρας RS-232, ως αρχείο περιφερειακής συσκευής, δηλ. TD = "/dev/ttyS0"

Κατά την τεχνική που χρησιμοποιεί ο Χαρβάτης (1990), η **OPENCNT** είναι υπορουτίνα σε γλώσσα FORTRAN που προσθέτει **NUL** στο τέλος της μεταβλητής character string **TD** και καλεί την **OpenCnt**. Η μεταβλητή **TD** του προγράμματος FORTRAN (δηλ. η **df** της **OpenCnt**) πρέπει, όπως έχει συζητηθεί, να είναι ένα **Null-terminated string** χαρακτήρων. Στην FORTRAN, αντίθετα από την C, οι μεταβλητές χαρακτήρων δεν έχουν τον χαρακτήρα **NUL** στο τέλος. Αυτό έχει σαν αποτέλεσμα, η **OpenCnt** να μην μπορεί να καλείται ποτέ με επιτυχία απ' ευθείας από πρόγραμμα FORTRAN αλλά πρέπει να καλείται πάντα μέσω μιας ενδιάμεσης υπορουτίνας π.χ. της **OPENCNT,** που προσθέτει το απαραίτητο **NUL** και καλεί την **OpenCnt**. Η υπορουτίνα αυτή περιέχεται στο αρχείο **onion.F** και έχει ως εξής:

```
C     Dimitris Harvatis – March 90
C     George Alexopoulos – April 08
#define orOPENCNT(i,name0) OpenCnt(%ref(i),%ref(name0))
C     This subroutine is called by the FORTRAN
C     programs instead of the OpenCnt
C     subroutine.
C     A null character is concatenated to the character
C     argument and the original subroutine is called.
C     However, the character argument that is to be
C     passed to the original routine, must not be
C     longer than 99 characters.
      subroutine OPENCNT(i,name)
      character*(*) name
      integer*4 i,l
      character*100 name0
      l=len_trim(name)+1
      name0=name
      name0(l:l)=char(0)
      call orOPENCNT(i,name0)
      return
      end
```

και η λειτουργία της αναλύεται παρακάτω

Σύνταξη: **subroutine OPENCNT(i,name)**

integer*4 I

character*(*) name

Περιγραφή: Η υπορουτίνα **OPENCNT** είναι γραμμένη σε FORTRAN. Το μόνο που κάνει είναι να προσθέτει ένα χαρακτήρα **NUL (0)** στο τέλος της παραμέτρου name και να καλεί την OpenCnt.

Μεταβλητές Εισόδου: **character*(*) name**

Είναι μια μεταβλητή χαρακτήρων. Το μήκος της ορίζεται στο πρόγραμμα που καλεί την **OPENCNT**. Τα περιεχόμενα της μεταβλητής **name** αντιγράφονται στη μεταβλητή **name0**. Στην **name0**, μετά τους χαρακτήρες της **name** γράφεται ένας χαρακτήρας **NUL (0)**. Η **name0** που περνιέται στην **OpenCnt**, έχει μήκος 100 χαρακτήρες άρα το μήκος της **name** δεν πρέπει να ξεπερνά τους 99 χαρακτήρες.

Μεταβλητές Εξόδου: **integer*4 i**

character*100 name0

Η **name0** χρησιμοποιείται όπως περιγράφθηκε. Ο ακέραιος **i** περνιέται αυτούσιος στην **OpenCnt**.

Σε σχέση με την προγραμματιστική υλοποίηση Χαρβάτη (1990), η **OPENCNT**, για να λειτουργήσει στα πλαίσια του Λ/Σ που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

- Χρήση κατάλληλης εντολής **#define**
- Χρήση της συνάρτησης βιβλιοθήκης **len_trim** αντί της **len** (βλ. και **info g77**)

Κλήση της **SetCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **SetCnt**, π.χ. με το FORTRAN όνομα SETCNT, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define SETCNT(IDV,T) SetCnt(%val(IDV),%val(T))

καθόσον και οι δύο μεταβλητές πρέπει να μεταφέρονται "by value",
και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL SETCNT(IDV, T)

Κλήση της **ClearCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **ClearCnt**, π.χ. με το FORTRAN όνομα CLEARCNT, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define CLEARCNT(IDV) ClearCnt(%val(IDV))

καθόσον η μοναδική μεταβλητή πρέπει να μεταφέρεται "by value"
και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL CLEARCNT(IDV)

Κλήση της **CloseCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **CloseCnt**, π.χ. με το FORTRAN όνομα CLOSECNT, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define CLOSECNT(IDV) CloseCnt(%val(IDV))

καθόσον η μοναδική μεταβλητή πρέπει να μεταφέρεται "by value"
και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL CLOSECNT(IDV)

Κλήση της **ToCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **ToCnt**, π.χ. με το FORTRAN όνομα TOCNT, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define TOCNT(IDV,CHI) ToCnt(%val(IDV),%ref(CHI))

καθόσον η πρώτη μεταβλητή πρέπει να μεταφέρεται "by value", η δεύτερη "by reference" και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL TOCNT(IDV,CHI)

Κλήση της **FromCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **FromCnt**, π.χ. με το FORTRAN όνομα FMCNT, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define FMCNT(IDV,M) FromCnt(%val(IDV),%ref(M))

καθόσον η πρώτη μεταβλητή πρέπει να μεταφέρεται "by value", η δεύτερη "by reference" και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL FMCNT(IDV,M)

Κλήση της **DataCnt**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **DataCnt**, π.χ. με το FORTRAN όνομα DATAIN, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define DATAIN(IDV,IC,ICNT) DataCnt(%val(IDV),%ref(IC),%ref(ICNT))

καθόσον η πρώτη μεταβλητή πρέπει να μεταφέρεται "by value", η δεύτερη και η τρίτη "by reference" και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL DATAIN(IDV,IC,ICNT)

Κλήση της **Time_to_I**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **Time_to_I**, π.χ. με το FORTRAN όνομα ITIME, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define ITIME(ISEC,IMIN,IHRS,JD,JM,JY)
Time_to_I(%ref(ISEC),%ref(IMIN),%ref(IHRS),%ref(JD),%ref(JM),%ref(JY))

καθόσον όλες οι μεταβλητές πρέπει να μεταφέρονται "by reference" και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL ITIME(ISEC,IMIN,IHRS,JD,JM,JY)

Κλήση της **Ellapsed**

Αν από το πρόγραμμα FORTRAN χρειάζεται να κληθεί η **Ellapsed**, π.χ. με το FORTRAN όνομα SECNDS, τότε στην αρχή του προγράμματος FORTRAN, πρέπει να τοποθετηθεί η εντολή προεπεξεργασίας:

#define SECNDS(TSTART,TICS,TWAIT,TICS2)
Ellapsed(%ref(TSTART),%ref(TICS),%ref(TWAIT),%ref(TICS2))

καθόσον όλες οι μεταβλητές πρέπει να μεταφέρονται "by reference" και μέσα στο πρόγραμμα θα υπάρχει η εντολή:

CALL SECNDS(TSTART,TICS,TWAIT,TICS2)

Αν στο FORTRAN πρόγραμμα του χρήστη, υπάρχει σκοπός να χρησιμοποιούνται όλες ή έστω πολλές από τις πιοπάνω υπορουτίνες, είναι δυνατόν οι εντολές **#define**, να τοποθετηθούν σε δικό τους ξεχωριστό αρχείο. Για τις ανάγκες της παρούσης ΔΕ κατασκευάσθηκε το αρχείο **count.f.i** με το εξής περιεχόμενο:

```
C George Alexopoulos – April 08
#define ITIME(ISEC,IMIN,IHRS,JD,JM,JY) Time_to_I(%ref(ISEC),%ref(IMIN),%ref(IHRS),%ref(JD),%ref(JM),%ref(JY))
#define FMCNT(IDV,M) FromCnt(%val(IDV),%ref(M))
#define TOCNT(IDV,CHI) ToCnt(%val(IDV),%ref(CHI))
#define SETCNT(IDV,T) SetCnt(%val(IDV),%val(T))
#define CLEARCNT(IDV) ClearCnt(%val(IDV))
#define SECNDS(TSTART,TICS,TWAIT,TICS2) Ellapsed(%ref(TSTART),%ref(TICS),%ref(TWAIT),%ref(TICS2))
#define DATAIN(IDV,IC,ICNT) DataCnt(%val(IDV),%ref(IC),%ref(ICNT))
#define CLOSECNT(IDV) CloseCnt(%val(IDV))
```

Σε τέτοια περίπτωση τα προγράμματα FORTRAN, που καλούν τις υπόψιν υπορουτίνες πρέπει στην αρχή να έχουν την εντολή προεπεξεργασίας:

#include "count.f.i"

Εννοείται ότι για την ορθή λειτουργία της εντολής **#include** πρέπει τόσο αυτά τα προγράμματα FORTRAN, όσο και το αρχείο **count.f.i**, να βρίσκονται στο ίδιο directory. Διαφορετικά, στην εντολή #include, θα πρέπει να ορίζεται το σχετικό path στο οποίο βρίσκεται το αρχείο **count.f.i**.

4.8.3 Εντολές μετάφρασης

Για τη μετάφραση προγράμματος σε γλώσσα προγραμματισμού C, χρησιμοποιείται η εντολή:

$gcc –c –g *file*.c –o *file*.o

όπου

file.c το αρχείο που περιέχει τον source κώδικα και *file.o* το αρχείο που θα περιέχει τον object κώδικα

Σχετικά μπορεί κανείς να μελετήσει και το **man gcc**.

Για τη μετάφραση προγράμματος σε γλώσσα προγραμματισμού FORTRAN, χρησιμοποιείται η εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –c –g *file*.F–o *file*.o

όπου

file.F το αρχείο που περιέχει τον source κώδικα (το κεφαλαίο *F*, δηλώνει ότι το FORTRAN πρόγραμμα χρειάζεται preprocessing, όπως έχει εξηγηθεί) και *file.o* το αρχείο που θα περιέχει τον object κώδικα

Σχετικά μπορεί κανείς να μελετήσει και το **man g77**.

Σχόλια: Ο διακόπτης –ffixed-line-length-132 πρέπει να χρησιμοποιείται για να καλύψει την περίπτωση που κάποιες γραμμές εντολών του κώδικα ξεπερνούν σε μήκος την $72^{η}$ στήλη, που είναι και το εξ'ορισμού όριο μήκους γραμμής εντολής στην γλώσσα FORTRAN. Αυτό π.χ. μπορεί να συμβαίνει σε μία εντολή **#define**. Ο διακόπτης –fno-underscoring πρέπει να χρησιμοποιείται για να εμποδίσει την κατά τη

μετάφραση αυτόματη πρόσθεση του χαρακτήρα underscore σε ονόματα στοιχείων του προγράμματος που καθορίζονται από τον χρήστη. Εννοείται δηλαδή, ότι όλοι οι αναγκαίοι χαρακτήρες underscore έχουν ήδη τοποθετηθεί στα ονόματα αυτά από τον χρήστη. Ο διακόπτης –fcase-preserve πρέπει να χρησιμοποιείται για να επιτρέψει τη χρήση μικρών και κεφαλαίων χαρακτήρων στον source κώδικα FORTRAN. Εξ'ορισμού οι FORTRAN εντολές, είτε είναι γραμμένες με μικρά είτε με κεφαλαία, είναι ισοδύναμες. Με το διακόπτη –fcase-preserve γίνονται διαφορετικές. Αυτό π.χ. μπορεί να συμβαίνει σε μία εντολή **#define**.

4.8.4 Συγκρότηση βιβλιοθήκης

Οι υπορουτίνες σε γλώσσα προγραμματισμού C, και σε FORTRAN (open_fsub), όπως παρουσιάσθηκαν μπορούν να χρησιμοποιούνται από ένα άλλο πρόγραμμα, π.χ. σε γλώσσα προγραμματισμού FORTRAN, αλλά πρέπει ο χρήστης να γνωρίζει που βρίσκονται. Για να απαλλαχθεί ο χρήστης από αυτόν τον περιορισμό, με τις υπορουτίνες αυτές μπορεί να συγκροτηθεί μία βιβλιοθήκη, η οποία να τοποθετηθεί στην προδιαγεγραμμένη για αυτόν τον σκοπό θέση του Λ/Σ:

/usr/local/lib

Η συγκρότηση της βιβλιοθήκης με το όνομα **count** μπορεί να γίνει με το ακόλουθο *makefile* (βλ. και **man make**)

```
#George Alexopoulos & Dimitris Karangelos – April 08

libdir=/usr/local/lib/
count=$(libdir)libcount.a

$(count): chardv.o datadv.o opendv.o date.o modedv.o secs.o onion.o
	ar urv $(count) chardv.o datadv.o opendv.o date.o modedv.o secs.o onion.o
	chmod o+r $(count)

chardv.o: chardv.c counter.h
	gcc -c -g chardv.c -o chardv.o

datadv.o: datadv.c counter.h
	gcc -c -g datadv.c -o datadv.o

opendv.o: opendv.c counter.h
	gcc -c -g opendv.c -o opendv.o

date.o: date.c counter.h
	gcc -c -g date.c -o date.o

modedv.o: modedv.c counter.h
	gcc -c -g modedv.c -o modedv.o

secs.o: secs.c counter.h
	gcc -c -g secs.c -o secs.o

onion.o: onion.F
	g77 -fno-underscoring -ffixed-line-length-132 -fcase-preserve -c -g onion.F -o onion.o
```

όπου το αρχείο **counter.h** περιέχει δηλώσεις extern και έχει το ακόλουθο περιεχόμενο:

```
/* AUTHOR : D. Harvatis – March 88 */

extern void FromCnt();    /* FromCnt(dv,c)  int dv; char c; */
extern void ToCnt();      /* ToCnt(dv,c)    int dv; char c; */
extern void OpenCnt();    /* OpenCnt(dv,df) int *dv; char *df; */
extern void CloseCnt();   /* CloseCnt(dv)   int dv; */
extern void ClearCnt();   /* ClearCnt(dv)   int dv; */
extern void DataCnt();    /* DataCnt(dv,err,n)    int dv; int *err; long *n; */
extern void SetCnt();     /* SetCnt(dv,t)   int dv,t; */
extern void Time_to_I();  /* Time_to_I(s,m,h,D,M,Y)     int *s,*m,*h,*D,*M,*Y; */
extern void Ellapsed();   /* Ellapsed(sref,tref,s,t) long *sref,*terf,*s,*t; */
```

Μετά τη συγκρότηση της βιβλιοθήκης count, ο χρήστης μπορεί να καλέσει, π.χ. από ένα πρόγραμμα σε FORTRAN, οποιαδήποτε υπορουτίνα από αυτές που περιέχονται σε αυτήν και να τη χρησιμοποιήσει στο εκτελέσιμό του. Σε αυτήν την περίπτωση για την μετάφραση και την παραγωγή του εκτελέσιμου, πρέπει να χρησιμοποιήσει την εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –g *file*.F –l count –o *file*

όπου

file.F το αρχείο που περιέχει τον source κώδικα (το κεφαλαίο *F*, δηλώνει ότι το FORTRAN πρόγραμμα χρειάζεται preprocessing, όπως έχει εξηγηθεί) και *file* το αρχείο που θα περιέχει το εκτελέσιμο.

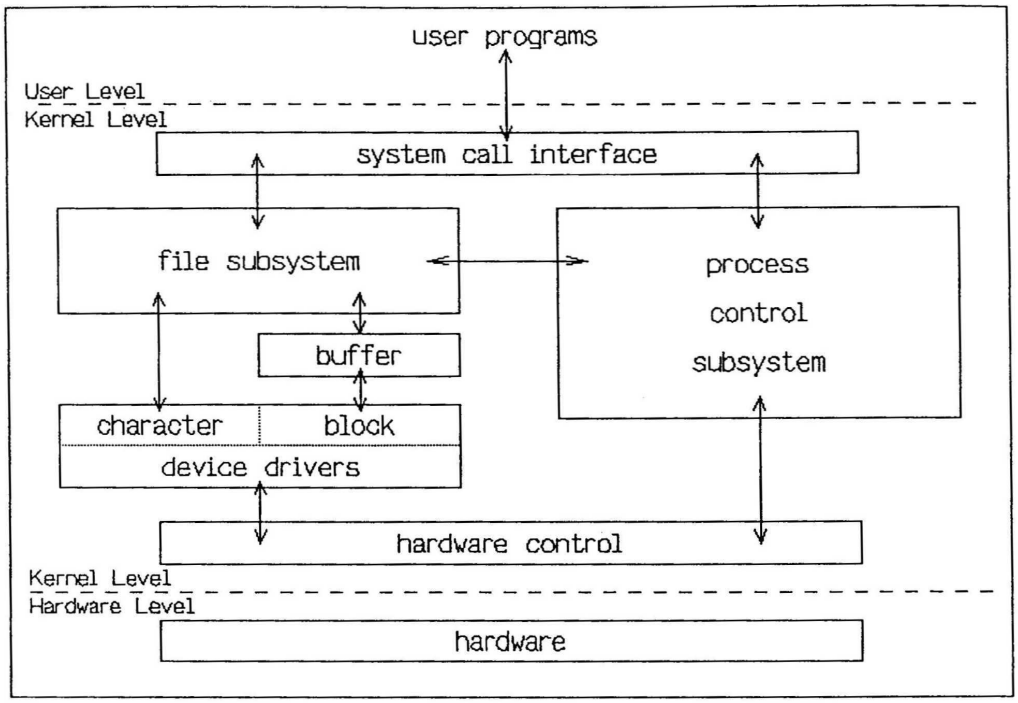
Σημειώνεται ότι τα αρχεία που παρουσιάζονται στην εδώ § 4.8 περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία.

4.9 Συμπεράσματα - Σχόλια

Στο Κεφάλαιο αυτό περιγράφθηκε με λεπτομέρεια το λογισμικό ρύθμισης, αποστολής και λήψης δεδομένων για σειριακή θύρα, όπως μετατράπηκε για το Λ/Σ LINUX χρησιμοποιώντας ως βάση το αντίστοιχο λογισμικό για το Λ/Σ UNIX / System V που δημιούργησε ο Χαρβάτης (1990). Πέρα από μερικές λίγες και αναπόφευκτες αλλαγές, διαπιστώθηκε ότι γενικά ο τρόπος λειτουργίας του λογισμικού δεν μεταβλήθηκε από τις μετατροπές. Αυτό σημαίνει ότι, παρόλη την μεγάλη χρονική απόσταση ηλικίας που χωρίζει το Λ/Σ UNIX / System V HP-UX 6.x και 7.x και τα σημερινά Λ/Σ LINUX (~20 έτη), υπάρχει σχετική "φορητότητα" (στα αγγλικά portability) του

λογισμικού από λειτουργικό σύστημα σε λειτουργικό σύστημα. Η φορητότητα αυτή διαπιστώθηκε και ερευνήθηκε με επιτυχία και από τον Καράγγελο (2008), κατά την μεταφορά λογισμικού από το HP-UX 6.x και 7.x σε Λ/Σ LINUX, για τις ανάγκες συγκρότησης νέου εξυπηρετητή –γ φασματοσκοπίας για το ΕΠΤ-ΕΜΠ.. Στο επόμενο 5^{ο} Κεφάλαιο, παρουσιάζεται ο τρόπος με τον οποίο έγινε η χρήση του μετατραπέντος λογισμικού για τη σύνδεση σειριακών μονάδων επικοινωνίας CANBERRA σε θύρα RS-232 Η/Υ.

ΔΙΑΓΡΑΜΜΑΤΑ ΤΟΥ 4ου ΚΕΦΑΛΑΙΟΥ



Διάγραμμα 4.1

Δομή του πυρήνα του Λ/Σ UNIX (Χαρβάτης, 1990)

ΚΕΦΑΛΑΙΟ 5
ΣΥΝΔΕΣΗ ΣΕΙΡΙΑΚΗΣ ΘΥΡΑΣ RS-232 Η/Υ ΜΕ ΣΕΙΡΙΑΚΕΣ ΣΥΣΚΕΥΕΣ CANBERRA

5.1 Εισαγωγή

Στο Κεφάλαιο αυτό θα ελεγχθεί και θα χρησιμοποιηθεί το λογισμικό ρύθμισης, αποστολής και λήψης δεδομένων για σειριακή θύρα, όπως μετατράπηκε για το Λ/Σ LINUX και παρουσιάσθηκε αναλυτικά στο προηγούμενο 4^{ο} Κεφάλαιο. Οι εφαρμογές για τις οποίες θα γίνει ο σχετικός έλεγχος και η χρήση έχουν συνοπτικά και με σειρά ως εξής:

(α) Σύνδεση, έλεγχος και ανάγνωση καταμετρητή CANBERRA Model 1776 μέσω μονάδας επικοινωνίας CANBERRA Model 1788

(β) Σύνδεση, έλεγχος και ανάγνωση καταμετρητή CANBERRA Model 1776 μέσω μονάδας επικοινωνίας CANBERRA Model 2088.

(γ) Σύνδεση, έλεγχος και ανάγνωση καταμετρητή CANBERRA Model 2071A/w03 με ενσωματωμένη μονάδα επικοινωνίας CANBERRA Model 207x-03

και

(δ) Σύνδεση, έλεγχος και ανάγνωση καταμετρητή CANBERRA Model 2071Aw/02 μέσω μονάδας επικοινωνίας CANBERRA Model 2088

Για τις εφαρμογές αυτές τροποποιήθηκαν ή αναπτύχθηκαν προγράμματα εφαρμογής σε γλώσσα προγραμματισμού FORTRAN77.

5.2 Συγκρότηση πειραματικής διατάξεως

Για την τροφοδοσία των συσκευών χρησιμοποιήθηκε μονάδα υποδοχής και τροφοδοσίας NIM Bin (βλ. και στο 2^{ο} Κεφάλαιο), CANBERRA – Model 1400. Πρόκειται για σχετικά παλαιή μονάδα του ΕΠΤ-ΕΜΠ (1975, με σειριακό αριθμό 11753599), η οποία συντηρήθηκε πρόσφατα (2008) από τον Πετρόπουλος (2008). Η μονάδα NIM Bin, τοποθετήθηκε για ευκολία σε μικρό φορητό ικρίωμα (στα αγγλικά rig) με κατάλληλες σχάρες στερέωσης (βλ. και φωτογραφία Διαγράμματος 5.1). Προκειμένου, η πειραματική διάταξη να προσομοιάζει τις συνθήκες καθημερινής πράξης του ΕΠΤ-ΕΜΠ, δηλαδή τις:

(α) Μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI

και

(β) Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI

οι καταμετρητές και οι άλλες μονάδες της διάταξης θα έπρεπε να είναι τοποθετημένοι στην μονάδα υποδοχής και τροφοδοσίας NIM Bin, κατά το σχήμα που έχει ήδη παρουσιασθεί στο Διάγραμμα 2.10.

Παρόλαυτά, η ανάπτυξη της πειραματικής διάταξης σε τέτοια έκταση δεν είναι σκόπιμη διότι θα έπρεπε να απασχολεί τα εξής στοιχεία:

(1) Ραδιενεργό πηγή

(2) Ανιχνευτή

(3) Τροφοδοτικό υψηλής τάσης

(4) Προενισχυτή

(5) Ενισχυτή, και

(6) Διαχωριστή (μονοκαναλικό αναλυτή)

τα οποία, δεν είναι γενικά διαθέσιμα στο ΕΠΤ-ΕΜΠ, καθόσον είτε χρησιμοποιούνται σε λειτουργούσες διατάξεις είτε φυλάσσονται για μελλοντική αντικατάσταση μονάδων που θα παρουσιάσουν βλάβη και δεν προσφέρονται για πειράματα.

Για το λόγο αυτό, στη θέση των στοιχείων (1) – (6) προτιμήθηκε να χρησιμοποιηθεί ένας "παλμοδότης τυχαίου θορύβου" (στα αγγλικά Random Noise Pulser) και συγκεκριμένα το Μοντέλο 1407R της CANBERRA (βλ. CANBERRA, 79-80, σελ. 220). Τα κύρια χαρακτηριστικά του υπόψιν παλμοδότη είναι ότι

"…παρέχει τυχαίους παλμούς σταθερού ρυθμιζόμενου ύψους με μεταβαλλόμενη μέση συχνότητα …"

Ο παλμοδότης αυτός τοποθετήθηκε στη θέση που σχηματικά δείχνει η φωτογραφία του Διαγράμματος 5.2 και ρυθμίσθηκε από το ποτενσιόμετρο "COUNT RATE" στην τιμή 2.5. Με ομοαξονικό καλώδιο σήματος τεχνολογίας BNC, μπορεί να συνδέεται η έξοδος TRIGGER OUT με την κατάλληλη είσοδο του καταμετρητή που είναι μέρος της πειραματικής διάταξης. Η έξοδος TRIGGER OUT δίνει θετικούς λογικούς παλμούς, ύψους ~ 6 V. Οι υπόλοιποι διακόπτες και ρυθμίσεις της συσκευής αυτής δεν

ενδιαφέρουν στην περιγραφόμενη σύνδεση (για τα λοιπά χαρακτηριστικά του CANBERRA – Model 1407R, βλ. και Παράρτημα).

Επισημαίνεται ότι, ο παλμοδότης αυτός δεν διαθέτει διακόπτη "ON-OFF", με αποτέλεσμα, όσο λειτουργεί η NIM Bin, να λειτουργεί και αυτός, πράγμα το οποίο είναι κακό για τα ηλεκτρονικά του συστήματος παλμοδότησης. Για το λόγο αυτό ο παλμοδότης CANBERRA – Model 1407R, "συρταρώνεται" στη μονάδα NIM Bin μόνο όταν χρειάζεται να χρησιμοποιηθεί.

Το ικρίωμα με τη μονάδα NIM Bin και τον παλμοδότη, τοποθετήθηκε στη γειτονιά δύο Η/Υ τύπου PC συμβατών, εκ των οποίων ο πρώτος είναι ο βασικός Η/Υ αυτής της ΔΕ, και ο δεύτερος είναι ο υπολογιστής "υποκλοπής" ή "spy PC" όπως αυτοί παρουσιάσθηκαν στο 3^{o} Κεφάλαιο. Η τοποθέτηση έγινε με τέτοιο τρόπο ώστε να υλοποιηθούν οι σειριακές συνδέσεις που αναλύονται στην παράγραφο 3.6 και στα Διαγράμματα 3.5 έως και 3.7 (συμπεριλαμβανομένου και του καλωδίου "υποκλοπής"). Τα καλώδια που χρησιμοποιούνται είναι αρκετά κοντά σε μήκος ώστε να μην δημιουργούνται παρεμβολές. Οι φωτογραφίες των Διαγραμμάτων 5.3 έως και 5.6 απεικονίζουν το συγκροτημένο σύνολο της πειραματικής διατάξεως. Η σειριακή σύνδεση ξεκινάει από τον βασικό Η/Υ (DB9 - DTE, βλ. και φωτογραφία του Διαγράμματος 5.4) και καταλήγει σε αρσενικό βύσμα DB25, κατάλληλο για να συνδεθεί στην θηλυκή υποδοχή DB25 της σειριακής συσκευής με την οποία χρειάζεται να αποκατασταθεί επικοινωνία (βλ. και φωτογραφία του Διαγράμματος 5.6).

5.3 Μονάδα επικοινωνίας 1788

5.3.1 Γενικά

Η μονάδα επικοινωνίας CANBERRA 1788 που θα χρησιμοποιηθεί έχει σειριακό αριθμό 1076280 και διακρίνεται στις φωτογραφίες των Διαγραμμάτων 5.7 (εμπρόσθια όψη) και 5.8 (οπίσθια όψη). Στη συνέχεια παρουσιάζονται τα εξωτερικά λειτουργικά μέρη της μονάδας που είναι απαραίτητα για την παρούσα ΔΕ. Τα λοιπά λειτουργικά μέρη αγνοούνται. Έτσι λοιπόν:

Στο εμπρόσθιο μέρος υπάρχουν

- ένα button που φέρει το όνομα FLAG και υπό κανονικές συνθήκες όταν χρησιμοποιείται από τον χρήστη, η μονάδα αποστέλλει στη σειριακή θύρα του Η/Υ τον χαρακτήρα (").

- ένας προειδοποιητικός λαμπτήρας με την ένδειξη ATTENTION ο οποίος ελέγχεται από τον Η/Υ και αναβοσβήνει σύμφωνα με κατάλληλες εντολές του χρήστη.

Στο πίσω μέρος υπάρχουν

- μία θύρα εισόδου 15 pin (CONTROL IN) η οποία συνδέει σε σειρά (Daisy Chain) την μονάδα με αντίστοιχο κατάλληλο καταμετρητή.

- μία θύρα εξόδου 15 pin (CONTROL OUT) η οποία συνδέει σε σειρά (Daisy Chain) την μονάδα με αντίστοιχο κατάλληλο καταμετρητή.

- μία θύρα, τύπου BNC, για τη μεταφορά παλμών από τον καταμετρητή προς την μονάδα {SIGNAL IN (#) }. Όταν στη θύρα αυτή καταφτάσει ένας παλμός +5V τότε αποστέλλεται στον Η/Υ ο χαρακτήρας(#).

- μία θύρα επικοινωνιών (EIA DATA) τύπου RS-232 female 25 pin (η μονάδα χρησιμοποιείται ως εξοπλισμός DTE).

5.3.2 Ρύθμιση

Υπενθυμίζεται ότι, κατά την κατάστρωση του λογισμικού, οι υπορουτίνες ρυθμίσεως της σειριακής θύρας RS-232 του Η/Υ, γράφθηκαν ώστε να τηρούνται οι ακόλουθες προδιαγραφές:

Ταχύτητα Μετάδοσης 9600 baud.

Μήκος χαρακτήρα 7 bits.

Parity EVEN.

Ένα stop bit

Όπως πληροφορεί το εγχειρίδιο της συσκευής, αυτή όπως έρχεται προρυθμισμένη από το εργοστάσιο κατασκευής έχει τους εσωτερικούς "βραχυκυκλωτήρες" (στα αγγλικά jumbers) ρύθμισης στις θέσεις:

1) Βραχυκυκλωτήρας Ext 96 (= ταχύτητα μετάδοσης 9600 baud)
2) Βραχυκυκλωτήρας O (parity) EVEN
3) Βραχυκυκλωτήρας P (Parity Control) ON
4) Βραχυκυκλωτήρες C1, C2, C3: Βραχυκυκλωμένοι (in)
5) Βραχυκυκλωτήρες M1, M2, M3: Μη Βραχυκυκλωμένοι (out)

6) Βραχυκυκλωτήρας 1S (αριθμός stop bits) Ένα stop bit

7) Βραχυκυκλωτήρας #L: Βραχυκυκλωμένος (in)

8) Βραχυκυκλωτήρας (#, non #): #

9) Βραχυκυκλωτήρας (X, non X): X

10) Βραχυκυκλωτήρας XL: Βραχυκυκλωμένος (in)

Προφανώς, πριν την σύνδεση της συσκευής έγινε ο απαραίτητος έλεγχος των βραχυκυκλωτήρων ώστε αυτή να είναι συμβατή με τις ρυθμίσεις της σειριακής θύρας RS-232 του Η/Υ. Για το σκοπό αυτό ανοίχθηκε το εσωτερικό της συσκευής και ελέγχθηκαν οι ορθές θέσεις των "βραχυκυκλωτήρων" και βρέθηκαν τα ακόλουθα:

1) Βραχυκυκλωτήρας Ext 96 (= ταχύτητα μετάδοσης 9600 baud)

2) Βραχυκυκλωτήρας O (parity) EVEN

3) Βραχυκυκλωτήρας P (Parity Control) ON

4) Βραχυκυκλωτήρες C1 (out), M1 (in)

5) Βραχυκυκλωτήρες C2 (out), M2 (in)

6) Βραχυκυκλωτήρες C3 (in), M3 (out), αλλά η ρύθμισή τους είναι αδιάφορη για την καλωδίωση που χρησιμοποιείται

7) Βραχυκυκλωτήρας 1S (αριθμός stop bits) Ένα stop bit

8) Βραχυκυκλωτήρας #L: Βραχυκυκλωμένος (in)

9) Βραχυκυκλωτήρας (#, non #): #

10) Βραχυκυκλωτήρας (X, non X): X

11) Βραχυκυκλωτήρας XL: Βραχυκυκλωμένος (in)

12) Βραχυκυκλωτήρας N: Βραχυκυκλωμένος (in)

Οι ρυθμίσεις που βρέθηκαν είναι λίγο διαφορετικές από τις εργοστασιακές προρυθμίσεις. Η διαφορά εξηγείται δεδομένου ότι η μονάδα επικοινωνίας 1788, βρίσκεται στο ΕΠΤ-ΕΜΠ πάνω 15 έτη και σε αυτό το διάστημα, μέχρι σήμερα, πιθανόν να έχει ρυθμισθεί και για άλλες εργασίες, διαφορετικές από τους σκοπούς της παρούσης ΔΕ. Οι ρυθμίσεις που βρέθηκαν, δεν μεταβλήθηκαν διότι δεν φαίνεται ότι κάποια από αυτές επηρεάζει την επιδιωκόμενη σειριακή επικοινωνία. Εξάλλου η

μονάδα επικοινωνίας 1788, όπως σημειώθηκε χρησιμοποιείται όπως έχει στο ΕΠΤ-ΕΜΠ για την εφαρμογή "Μετρήσεις συγκέντρωσης ^{137}Cs σε στερεά δείγματα με ανιχνευτή NaI" στην οποία όπως έχει καταγραφεί και στο 2^{o} Κεφάλαιο, συνδέεται με την σειριακή θύρα ενός Η/Υ PC συμβατού με ρυθμίσεις Ταχύτητα Μετάδοσης 9600 baud, Μήκος χαρακτήρα 7 bits, Parity EVEN, Ένα stop bit [βλ. και Βοσυνιώτης (1990)]. Σημειώνεται ότι η μονάδα επικοινωνίας 1788 είναι κατασκευασμένη έτσι ώστε να "διαβάζει" ψηφιακούς καταχωρητές από αντίστοιχους κατάλληλους καταμετρητές μήκους έξι χαρακτήρων.

5.3.3 Πρωτόκολλο επικοινωνίας

Η μονάδα επικοινωνίας διαθέτει πρωτόκολλο σειριακής σύνδεσης μέσω του οποίου η σειριακή θύρα του συνδεδεμένου Η/Υ είναι δυνατόν να ελέγχει έναν ή περισσότερους καταμετρητές και να παραλαμβάνει δεδομένα από αυτούς. Το πρωτόκολλο αυτό έχει δύο συνιστώσες, το πρωτόκολλο αποστολής και το πρωτόκολλο λήψης, ως εξής: (α) Αποστολή χαρακτήρων ASCII ≡ Εντολές ελέγχου από τον Η/Υ προς τη μονάδα επικοινωνίας 1788, και (β) Λήψη χαρακτήρων ASCII ≡ Πληροφορίες από την μονάδα επικοινωνίας προς τον Η/Υ. Τα δύο πρωτόκολλα είναι (όπως δίνονται από το αντίστοιχο εγχειρίδιο, βλ. και Παράρτημα):

Πρωτόκολλο αποστολής από τον Η/Υ

ASCII Code (octal)	**Χαρακτήρας**	**Ορισμός**
122	R	Will reset the system
101	A	Will turn on the F.P[1] light.
123	S	Will start the system
102	B	Will turn off the F.P. light
110	H	Will halt the system
130	X	Will give the X output pulse (όταν ο βραχυκυκλωτήρας N είναι in – ο παλμός εξόδου X, δεν χρησιμοποιείται στην εδώ εφαρμογή)

[1] Front Panel

111	I	Will cause the Data Acquisition modules to go into their Readout mode (if they are not already in the Readout mode) and the next unit in order will read out its six characters followed by a space (ASCII Code 040)

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που δέχεται η μονάδα 1788 από τον Η/Υ ερμηνεύονται ως εντολές που αναλύονται ως εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
82	R	επαναφέρει τις μετρητικές διατάξεις στην αρχική τους κατάσταση μηδενίζοντας τις ενδείξεις (reset).
65	A	ανάβει η ενδεικτική λυχνία.
83	S	δίνει εντολή στις μετρητικές διατάξεις να αρχίσουν μία μέτρηση (start)
66	B	σβήνει η ενδεικτική λυχνία.
72	H	δίνει εντολή στις μετρητικές διατάξεις να σταματήσουν τη μέτρηση (halt)
73	I	δίνει εντολή στη συσκευή ώστε να ενεργοποιήσει τον λογικά "επόμενο κατά σειρά" scaler ώστε αυτός να στείλει τους έξι χαρακτήρες που συγκροτούν την ένδειξη του, ακολουθούμενους από ένα χαρακτήρα SPACE , στον Η/Υ.

Πρωτόκολλο λήψης προς τον Η/Υ

ASCII Code (octal)	**Χαρακτήρας**	**Ορισμός**
040	Space	Follows data from modules
041	!	The system has just stopped or there are no more scalers to read.

042	"	Given from pushing the FLAG button.
043	#	Given from an input to Sig. In (#)
060 έως 067	0 έως 7	Data from modules
070, 071	8, 9	Data from modules

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που αποστέλλει η μονάδα 1788 στον τον Η/Υ είναι πληροφορίες που σημαίνουν τα εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
32	Space	Τερματικός χαρακτήρας μίας εξαψήφιας ένδειξης
34	"	Πατήθηκε το κουμπί FLAG.
35	#	Παράγεται από την μονάδα 1788 αν αυτή δεχθεί ένα παλμό τάσης +5V στην είσοδο SIGNAL IN. Ένας τέτοιος παλμός παράγεται π.χ. από την έξοδο GATE OUT ενός καταμετρητή CANBERRA - Model 1776. Στη συγκρότηση που εξετάζεται στην επόμενη παράγραφο, ο καταμετρητής 1776 (έξοδος GATE OUT) είναι πάντα συνδεδεμένος με τη μονάδα 1788 (είσοδος SIGNAL IN) με κατάλληλο BNC καλώδιο.
48 έως 57	0 έως 9	Ψηφία που κατά περίπτωση συγκροτούν το περιεχόμενο της ένδειξης
33	!	Ο χαρακτήρας αυτός σημαίνει DONE και αποστέλλεται στις εξής περιπτώσεις: - όταν η λειτουργία των καταμετρητών (counters) ολοκληρώνεται εξαιτίας

συμπληρώσεως του προκαθορισμένου χρόνου (preset time).

- όταν η λειτουργία ενός καταμετρητή διακόπτεται λόγω πιέσεως του πλήκτρου STOP επί του καταμετρητή
- όταν η λειτουργία των καταμετρητών διακόπτεται λόγω αποστολής από τον Η/Υ χαρακτήρα Η (halt).
- όταν έχει αρχίσει η διαδικασία μεταφοράς των περιεχομένων των καταμετρητών στον Η/Υ και έχει αναγνωσθεί και ο τελευταίος "λογικά" καταμετρητής.

Έτσι, σύμφωνα με το εγχειρίδιο της μονάδας 1788 (βλ. και Παράρτημα), την πιοπάνω ανάλυση και τα όσα σημειώνει ο Χαρβάτης (1990) μια τυπική συνδιάλεξη μεταξύ Η/Υ και 1788 μπορεί να έχει την ακόλουθη μορφή:

Η/Υ	**1788**	**Σχόλια**
R		Μηδενισμός των ενδείξεων.
S		Οι καταμετρητές αρχίζουν μια μέτρηση.
H		Οι καταμετρητές σταματούν. Η εντολή αυτή μπορεί να παραλειφθεί αν αφεθούν οι μετρήσεις να ολοκληρωθούν λόγω προκαθορισμένου χρόνου.
	!	Οι μετρήσεις σταμάτησαν ή ολοκληρώθηκαν.
I		Να σταλεί η ένδειξη του πρώτου scaler στον Η/Υ.
	6 ΨΗΦΙΑ	Τα έξι ψηφία της ένδειξης του πρώτου scaler αποστέλλονται στον Η/Υ.
	ΚΕΝΟ	Η μετάδοση της ένδειξης του πρώτου scaler

		ολοκληρώθηκε.
I		Να σταλεί η ένδειξη του επόμενου scaler στον H/Y.
	6 ΨΗΦΙΑ	Τα έξι ψηφία της ένδειξης του επόμενου scaler αποστέλλονται στον H/Y.
	ΚΕΝΟ	Η μετάδοση της ένδειξης του επόμενου scaler ολοκληρώθηκε.
I		Να σταλεί η ένδειξη του επόμενου scaler στον H/Y.
	!	Δεν υπάρχει άλλος scaler συνδεδεμένος.

Σημειώνεται σε αυτό το σημείο ότι η μονάδα 1788 δεν μπορεί να χρησιμοποιήσει το πρωτόκολλο XON – XOFF για επικοινωνία με την σειριακή θύρα ενός H/Y, διότι αυτοί οι χαρακτήρες δεν αναγνωρίζονται αν σταλούν από τον H/Y προς τη σειριακή θύρα της μονάδας 1788. Επισημαίνεται βέβαια ότι στις υπορουτίνες ρύθμισης της σειριακής θύρας RS-232 του H/Y, που συμπεριλαμβάνονται στην βιβλιοθήκη count, το πρωτόκολλο XON – XOFF είναι απενεργοποιημένο.

5.3.4 Σύνδεση με την πειραματική διάταξη

Μετά τη ρύθμιση η υπόψιν μονάδα επικοινωνίας τοποθετήθηκε στην μονάδα NIM Bin μαζί με τον καταμετρητή CANBERRA – Model 1776. Ο καταμετρητής συνδέθηκε με τη μονάδα επικοινωνίας με καλωδίωση Daisy Chain, με τον τρόπο που διακρίνεται στην φωτογραφία του Διαγράμματος 5.9. Η μονάδα επικοινωνίας συνδέθηκε με το αρσενικό βύσμα DB25 σε αναμονή, το οποίο διακρίνεται στη φωτογραφία του Διαγράμματος 5.6. Η σύνδεση όπως έγινε διακρίνεται στη φωτογραφία του Διαγράμματος 5.10.

5.3.5 Προγραμματιστική υλοποίηση από τον Δ. Χαρβάτη (1990)

Ένα ολοκληρωμένο παράδειγμα χρησιμοποίησης των υπορουτινών της βιβλιοθήκης **count,** όπως αυτή αναλύεται στο κείμενο της ΔΕ του Χαρβάτη (1990), για συνεργασία με μονάδα επικοινωνίας 1788, προκύπτει από τη μετασκευή του προγράμματος FORTRAN **RRADON** (Σιμόπουλος, 1986), ώστε να λειτουργεί στο Λ/Σ HP-UX 6.x και 7.x. Το πρόγραμμα αυτό ήταν σε παραγωγική χρήση στο ΕΠΤ-

ΕΜΠ με χρήση Η/Υ HP 9000/320, για τη συνεχή, επί εικοσιτετραώρου βάσεως, παρακολούθηση της συγκέντρωσης του ισοτόπου Bi-214 (609 keV) θυγατρικού του Rn-222 που βρίσκεται στην ατμόσφαιρα, στα πλαίσια της εφαρμογής "Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI", οι αρχές της οποίας παρουσιάσθηκαν στο 2^{ο} Κεφάλαιο. Τα σημεία του προγράμματος που πρέπει να αλλάξουν, προκειμένου αυτό να προσαρμοσθεί για περιβάλλον Λ/Σ LINUX / POSIX και γλώσσα προγραμματισμού FORTRAN77, σημαίνονται με κατάλληλη σκίαση [ ].

Περιγραφή: Στο ΕΠΤ-ΕΜΠ, η μονάδα 1788 μπορεί να συνδέεται συνήθως με έναν καταμετρητή/χρονόμετρο (counter/timer) CANBERRA - Model 1776 και έναν ή περισσότερους καταμετρητές (counters), κατά προτίμηση επίσης CANBERRA – Model 1776, χονδρικά κατά τη φωτογραφία του Διαγράμματος 2.14. Η διάταξη αυτή, που ονομάζεται διάταξη counter-timer, λειτουργεί ως εξής: Ο timer (το χρονόμετρο) ρυθμίζεται εκ των προτέρων για ένα συγκεκριμένο χρονικό διάστημα που είναι η διάρκεια της μέτρησης. Στο χρονικό διάστημα αυτό, ο counter (ο καταμετρητής) καταγράφει παλμούς που προέρχονται από κάποιον ανιχνευτή (π.χ. κρύσταλλο NaI ή Θάλαμο Geiger-Mueller). Όταν η ένδειξη του timer φτάσει την προκαθορισμένη τιμή η μέτρηση τελειώνει. Τότε ο counter/timer 1776, αποτυπώνει στο ένα του παράθυρο (scaler) το χρονικό διάστημα της μέτρησης και στο δεύτερο παράθυρο το πλήθος των παλμών που κατέγραψε κατά το διάστημα αυτό.

Το πρόγραμμα **RRADON** χρησιμοποιείται για τον έλεγχο των διατάξεων counter-timer και για τη συλλογή των μετρήσεών τους, τις οποίες αποθηκεύει σε αρχεία στον μαγνητικό δίσκο του υπολογιστή. Οι ακόλουθες παράμετροι, που διαβάζονται από ένα αρχείο π.χ. **RREXCO.DAT**, καθορίζουν την λειτουργία του προγράμματος **RRADON**:

NSC: Είναι ο αριθμός των μετρητικών διατάξεων που είναι συνδεδεμένες στη μονάδα 1788 (Number of SCalers). Πρέπει να είναι τουλάχιστον δύο (ένας timer και ένας counter). Η πρώτη λογική μονάδα είναι ο timer ενώ η δεύτερη είναι ο πρώτος counter κ.ε.

MEAS: Είναι το πλήθος των μετρήσεων από το οποίο σχηματίζεται κάθε δείγμα (μέγεθος δείγματος). Το πρόγραμμα **RRADON** υπολογίζει το μέσο

όρο και την τυπική απόκλιση των μετρήσεων κάθε δείγματος και τις γράφει σε ένα αρχείο εξόδου.

IDT: Είναι ο χρόνος μέτρησης. Η τιμή αυτή πρέπει να είναι ίδια με την τιμή στην οποία έχει ρυθμιστεί ο timer της διάταξης.

MMEAS: Μέγιστο μέγεθος δείγματος. Η παράμετρος αυτή δεν χρησιμοποιείται από το πρόγραμμα **RRADON**.

PDX: Μέγιστη αποδεκτή τιμή της τυπικής απόκλισης δείγματος. Δεν χρησιμοποιείται από το πρόγραμμα **RRADON**.

TSAMP: Χρονικό διάστημα που απέχει η έναρξη μέτρησης κάθε δείγματος από την έναρξη μέτρησης του προηγουμένου δείγματος (σε δευτερόλεπτα).

Οδηγίες Χρήσεως: Το πρόγραμμα **RRADON** τρέχει με την απλή εντολή :

rradon

Στην οθόνη εμφανίζεται το μήνυμα :

```
PLS PUT DAISY CHAIN ON LINE AND PRESS FLAG
To resume execution, type 'go'.
```

Ο χρήστης καλείται να αποκαταστήσει την συνδεσμολογία της μονάδας 1788 με τον Η/Υ και κατόπιν να πατήσει το διακόπτη flag (βρίσκεται πάνω στην μονάδα 1788). Στη συνέχεια πρέπει να πληκτρολογήσει την εντολή "go" οπόταν εμφανίζεται στην οθόνη το μήνυμα :

```
IF D/C IS ON LINE A MESSAGE FOLLOWS
IF NOT CPU LOOPS TILL D/C CONNECTED
IF D/C IS CONNECTED BUT CPU STILL LOOPS PLEASE PRESS FLAG
```

Αν το πρόγραμμα "κολλήσει" στο σημείο αυτό ο χρήστης πρέπει να πατήσει το flag. Ακολουθεί το μήνυμα:

```
DAISY CHAIN IS ON LINE - EXPERIMENT CAN GO AHEAD
Sample number : 1        Counting started
```

ενώ ακούγεται το χαρακτηριστικό "καμπανάκι" του τερματικού. Στο σημείο αυτό έχει αρχίσει η πρώτη μέτρηση του πρώτου δείγματος. Όταν ολοκληρωθούν όλες οι μετρήσεις του πρώτου δείγματος θα εμφανισθούν στην οθόνη ο αύξων αριθμός του δείγματος, το μέγεθος δείγματος, ο χρόνος μέτρησης, οι τιμές των μετρήσεων, ο μέσος όρος των τιμών αυτών, η τυπική τους απόκλιση, η ημερομηνία και η ώρα. Ο αύξων αριθμός του δείγματος, το μέγεθος δείγματος, ο μέσος όρος, η τυπική

απόκλιση, η ημερομηνία και η ώρα θα γραφούν επίσης και στο αρχείο εξόδου. Κατόπιν, το πρόγραμμα θα περιμένει μέχρι να έρθει η στιγμή να ληφθεί το δεύτερο δείγμα οπότε θα επαναληφθεί η προηγούμενη διαδικασία. Οποιαδήποτε στιγμή είναι δυνατόν να τερματισθεί το πρόγραμμα πατώντας δύο συνεχόμενες φορές το flag.

Αν κατά τη διάρκεια της μετάδοσης κάποιας μέτρησης σημειωθεί λάθος, τότε εμφανίζεται το μήνυμα :

ERROR DURING PREVIOUS RUN - RUN TO BE REPEATED

και η μέτρηση επαναλαμβάνεται.

Το Αρχείο Εξόδου: Το όνομα του αρχείου εξόδου καθορίζεται με βάση την ημερομηνία και την ώρα εκτελέσεως του προγράμματος. Συγκεκριμένα, έχει τη μορφή ενός εξαψήφιου αριθμού (α' μέρος) που ακολουθείται από μία τελεία και έναν τριψήφιο αριθμό (β' μέρος). Τα πρώτα τέσσερα ψηφία του α' μέρους είναι η ώρα (ώρες και λεπτά). Το πέμπτο και το έκτο είναι η ημερομηνία. Τα δύο πρώτα ψηφία του β' μέρους είναι ο μήνας ενώ το τελευταίο είναι το τελευταίο ψηφίο του έτους. Έτσι, το όνομα αρχείου **093114.030** αντιστοιχεί στην **14η Μαρτίου 1990, 9:31**.

Αν υπάρχει ήδη άλλο αρχείο με αυτό το όνομα, η ερώτηση :

File already existing - override ?

απευθύνεται στον χρήστη ο οποίος αν θέλει πράγματι να σβήσει το παλιό αρχείο πρέπει να απαντήσει **"YES"**, αλλιώς το πρόγραμμα τερματίζεται.

Η πρώτη γραμμή του αρχείου εξόδου περιέχει την ημερομηνία και την ώρα. Η δεύτερη περιέχει τον αριθμό των καταμετρητών που είναι συνδεδεμένοι στην μονάδα 1788, τον αριθμό των μετρήσεων ανά δείγμα και το χρόνο μέτρησης. Κάθε μία από τις επόμενες γραμμές του αρχείου αντιστοιχεί σε ένα δείγμα και περιέχει : τον αριθμό του δείγματος, το πλήθος των μετρήσεων, τον μέσο όρο και την τυπική απόκλιση των μετρήσεων του δείγματος, την ημερομηνία και την ώρα.

Κείμενο προγράμματος: Στη συνέχεια παρατίθενται το πρόγραμμα **RRADON** σε γλώσσα προγραμματισμού FORTRAN/9000, όπως αυτή υπήρχε στα συστήματα Η/Υ HP-9000/320-370 (βλ. και Hewlett Packard, 1987), που αναπτύχθηκε και χρησιμοποιήθηκε από τον Χαρβάτη (1990) για περιβάλλον Λ/Σ HP-UX 6.x και 7.x. Επειδή το κείμενο του προγράμματος προέκυψε στην παρούσα ΔΕ ως αποτέλεσμα σάρωσης και αναγνώρισης χαρακτήρων, είναι δυνατόν να υπάρχουν λάθη, παρόλο

που καταβλήθηκε προσπάθεια για τον περιορισμό τους. Επομένως, σε περίπτωση αμφιβολίας, ο αναγνώστης πρέπει να αναφέρεται στο πρωτότυπο κείμενο της ΔΕ του Δ. Χαρβάτη (1990). Το πρόγραμμα αποτελείται από ένα αρχείο **.f**, το οποίο περιέχει ένα κυρίως πρόγραμμα και μία υπορουτίνα. Τα σημεία του προγράμματος που πρέπει να αλλάξουν, προκειμένου αυτό να προσαρμοσθεί για περιβάλλον Λ/Σ LINUX / POSIX και γλώσσα προγραμματισμού FORTRAN77, σημαίνονται με κατάλληλη σκίαση [ ].

rradon.f

```
$INCLUDE 'count.f.i'
      PROGRAM RRADON
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.P.HARVATIS - April 88
      DIMENSION ICNT(3),COUNT(20,5)
      BYTE KSPACE,KDONE,M,KFLAG,YES,IY,BEL
      CHARACTER*10 FIL
      CHARACTER*17 STIME
      INTEGER*4 TICS,TICS2,TSTART,TWAIT,TSAMP
      DATA KSPACE,KDONE,KFLAG/1H ,1H!,1H"/,YES/'Y'/,BEL/07/
      PRINT 74
  74  FORMAT(/)
      CALL ITIME(ISEC,IMIN,IHRS,JD,JM,JY)
      JY1=JY-80
      ENCODE(10,111,FIL)IHRS,IMIN,JD,JM,JY1
 111  FORMAT(3I2,'.',I2,I1)
      CALL REPLACE(FIL,'0',' ')
      PRINT 114,FIL
      OPEN(UNIT=10,FILE=FIL,STATUS='OLD',ACCESS='SEQUENTIAL',ERR=100)
      PRINT 34
  34  FORMAT(/$'File already existing - override?')
      READ 35,IY
  35  FORMAT(A1)
      IF(IY.NE.YES) THEN
      STOP
      ENDIF
  36  CLOSE(UNIT=10,STATUS='DELETE',ERR=102)
 100  OPEN(UNIT=10,FILE=FIL,STATUS='NEW',ACCESS='SEQUENTIAL',ERR=101)
      OPEN(11,FILE='RREXCO.DAT')
      READ(11,37)NSC,MEAS,IDT,MMEAS,PDX,TSAMP
  37  FORMAT(4I5,F10.0,I5)
      DT=IDT
      NSN=1
      WRITE(10,53)JD,JM,JY,IHRS,IMIN,ISEC,NSC,MEAS,IDT
  53  FORMAT(I2,'-',I2,'-',I2,10X,I2,':',I2,':', I2,/3I5,2F10.3)
      ICC=0
      PAUSE ' PLS PUT DAISY CHAIN ON LINE AND PRESS FLAG'
      PRINT 1
   1  FORMAT(/'IF D/C IS ON LINE A MESSAGE FOLLOWS',
     1' IF NOT CPU LOOPS TILL D/C CONNECTED'/
     2'IF D/C IS CONNECTED BUT CPU STILL LOOPS PLEASE PRESS FLAG')
C     open RS-232 special file
      CALL OPENCNT(IDV,"/deV/ttyd05"//char(0))
C     wait until the flag button is pressed
   3  CALL FMCNT(IDV,M)
      IF(M.NE.KFLAG)GO TO 3
```

```
      PRINT 4
    4 FORMAT(/'DAISY CHAIN IS ON LINE - EXPERIMENT CAN GO AHEAD'/)
      CALL TOCNT(IDV,'A')
   40 SX=0
      SXX=0
      RN=0
      L1=MEAS
   85 DO 30 L=1,L1
C     reset counter
      CALL TOCNT(IDV,'R')
      IF(L.GT.1)GO TO 50
      IF(ISW.EQ.0)GO TO 151
C     set RS-232 driver for immediate responce
   51 CALL SETCNT(IDV,0)
C     read one character
   52 CALL FMCNT(IDV,M)
      IF(M.EQ.KFLAG)GO TO 45
   54 IF(M.EQ.0)GO TO 501
C     halt, reset and clear buffer
      CALL TOCNT(IDV,'H')
      CALL TOCNT(IDV,'R')
      CALL CLEARCNT(IDV)
C     get time since reference point
  501 CALL SECNDS(TSTART,TICS,TWAIT,TICS2)
      IF(TWAIT.LT.TSAMP)GO TO 52
C     get reference point
  151 CALL SECNDS(0,0,TSTART,TICS)
      ISW=1
      CALL ITIME(ISEC,IMIN,IHRS,JD,JM,JY)
      ENCODE(17,500,STIME)JD,JM,JY,IHRS,IMIN,ISEC
  500 FORMAT(I2,'/',I2,'/',I2,'-',I2,':',I2,':',I2)
      CALL REPLACE(STIME,'0',' ')
  141 PRINT 33,BEL,NSN
   33 FORMAT(/,A1,'Sample number : ',I4,5X,'Counting started')
C     start
   50 CALL TOCNT(IDV,'S')
C     set RS-232 driver for normal responce (wait mode)
      CALL SETCNT(IDV,1)
      CALL FMCNT(IDV,M)
C     led OFF
   14 CALL TOCNT(IDV,'B')
      CALL FMCNT(IDV,M)
      IF(M.EQ.KDONE)GO TO 44
C     halt
      CALL TOCNT(IDV,'H')
      CALL FMCNT(IDV,M)
      GO TO 44
C     check for double push on the flag button
   45 CALL SLEEP(2)
      CALL FMCNT(IDV,M)
      IF(M.EQ.KFLAG)GO TO 41
      GO TO 54
C     get data from NSC scalers
C     led ON
   44 CALL TOCNT(IDV,'A')
      DO 15 I=1,NSC
      CALL TOCNT(IDV,'I')
      CALL DATAIN(IDV,IC,ICNT)
      CALL FMCNT(IDV,M)
      IF(M.NE.KSPACE)ICC=1
      COUNT(L,I)=REAL(ICNT)
```

```
      IF(IC.NE.0)ICC=1
   15 CONTINUE
      CALL TOCNT(IDV,'I')
      CALL FMCNT(IDV,M)
      IF(COUNT(L,1).NE.DT.OR.M.NE.KDONE.OR.ICC.EQ.1)GO TO 18
      SXX=SXX+COUNT(L,2)**2
      SX=SX+COUNT(L,2)
      RN=RN+1.
      GOTO 30
C-----------------------------------------------------------------------
   18 PRINT 16,M,IC,ICC,COUNT(L,1),DT
   16 FORMAT(/'ERROR DURING PREVIOUS RUN - RUN TO BE REPEATED'
     1/' VALUES : ',' M= ',A1,'  IC=',I3,'  ICC=',I3,
     2' COUNT(1)=',F7.0,'  DT=',F7.0)
C     reset, start
      CALL TOCNT(IDV,'R')
      CALL TOCNT(IDV,'S')
      CALL FMCNT(IDV,M)
      GO TO 14
C-----------------------------------------------------------------------
   30 CONTINUE
      XM=SX/RN
      PSDX=100.*SQRT((SXX-RN*XM**2)/(RN*(RN-I.)))/XM
   84 PRINT 89,NSN,IFIX(RN),IDT,(IFIX(COUNT(L,2)),L=1,MEAS),XM,PSDX
     +,STIME
   89 FORMAT(1X,I4,2X,I4,2X,I4,3(2X,I5),4X,F6.1,2X,F5.2,' % ',A17)
   87 WRITE(10,38)NSN,IFIX(RN),XM,PSDX,STIME
   38 FORMAT(I5,1X,I5,1X,F6.1,F7.2,5X,A17)
      NSN=NSN+1
      GO TO 40
C     double push on the flag button stops the program
   41 CALL TOCNT(IDV,'B')
      CALL CLOSE(10)
      CALL CLOSE(11)
      PRINT 48,BEL
   48 FORMAT(A1,/'TWO FLAGS RECEIVED - END OF THIS EXPERIMENT'//)
      PRINT 114,FIL
  114 FORMAT('Output file name : ',A12/)
      STOP
C     errors
  101 PRINT 104
  104 FORMAT(/'File cannot be opened')
      STOP
  102 PRINT 105
  105 FORMAT(/'File cannot be overriden')
      STOP
      END
C=======================================================================
      SUBROUTINE REPLACE(STR,C1,C2)
      CHARACTER*(*) STR
      CHARACTER C1,C2
C     replaces C2 with Cl in STR
      DO 900 I = 1,LEN(STR)
      IF (STR(I:I) .EQ. C2) THEN
      STR(I:I) = C1
      ENDIF
  900 CONTINUE
      RETURN
      END
```

5.3.6 Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας

Στη συνέχεια παρατίθενται το πιο πάνω πρόγραμμα σε γλώσσα προγραμματισμού FORTRAN77, το οποίο βασίζεται στο αντίστοιχο του Χαρβάτη (1990). Το πρόγραμμα αποτελείται από ένα αρχείο **.f**, όπως στην προηγούμενη ενότητα. Πραγματοποιήθηκαν αλλαγές για την μετατροπή του από FORTRAN/9000 σε FORTRAN77. Οι διαφορές που προέκυψαν στο πρόγραμμα, μετά τη μετατροπή, δηλαδή οι προσθέσεις, αφαιρέσεις και αλλαγές σημαίνονται με κατάλληλη σκίαση [ ]. Το πρόγραμμα χρησιμοποιεί τις υπορουτίνες της βιβλιοθήκης **count,** οι οποίες παρουσιάσθηκαν στο προηγούμενο 4^{ο} Κεφάλαιο και λειτουργεί στο Λ/Σ LINUX / POSIX, έκδοση Fedora Core 8. Η καλή λειτουργία του προγράμματος δοκιμάσθηκε με επιτυχία σε αυτήν την ΔΕ στην πειραματική διάταξη της εδώ ενότητας (§ 5.3).

Το πρόγραμμα με το εδώ όνομα **RADON** μπορεί να χρησιμοποιείται –όπως και το προηγούμενο πρόγραμμα **RRADON**, για τον έλεγχο των διατάξεων counter-timer και για τη συλλογή των μετρήσεών τους, τις οποίες αποθηκεύει σε αρχεία. Η περιγραφή και οι οδηγίες χρήσεως του προγράμματος έχουν χονδρικά ακριβώς όπως στην προηγούμενη παράγραφο.

Σχόλια: Το πρόγραμμα **RADON** τρέχει με την απλή εντολή:

./radon

(αρκεί το αρχείο στο οποίο περιέχεται το προκύπτον εκτελέσιμο να έχει αυτό το όνομα και επίσης να έχει γίνει με ορθό τρόπο η μετάφραση και η σύνδεση).

Το όνομα του αρχείου εξόδου καθορίζεται με βάση την ημερομηνία και την ώρα εκτελέσεως του προγράμματος. Συγκεκριμένα, έχει τη μορφή ενός εξαψήφιου αριθμού (α' μέρος) που ακολουθείται από μία τελεία και έναν τετραψήφιο αριθμό (β' μέρος). Τα πρώτα τέσσερα ψηφία του α' μέρους είναι η ώρα (ώρες και λεπτά). Το πέμπτο και το έκτο είναι η ημερομηνία. Τα δύο πρώτα ψηφία του β' μέρους είναι ο μήνας ενώ τα δύο τελευταία είναι τα τελευταία ψηφία του έτους. Έτσι, το όνομα αρχείου **093114.0308** αντιστοιχεί στην **14η Μαρτίου 2008, 9:31**.
Αν υπάρχει ήδη άλλο αρχείο με αυτό το όνομα, η ερώτηση :

`File already existing – Press Y to override`

απευθύνεται στον χρήστη ο οποίος αν θέλει πράγματι να σβήσει το παλιό αρχείο πρέπει να απαντήσει **"Y"**, αλλιώς το πρόγραμμα τερματίζεται.

Κείμενο προγράμματος: Στη συνέχεια παρατίθενται το πρόγραμμα **RADON** σε γλώσσα προγραμματισμού FORTRAN77, όπως αυτή υπάρχει σε σύστημα Η/Υ τύπου PC συμβατού για περιβάλλον Λ/Σ LINUX / POSIX έκδοση Fedora Core 8. Το πρόγραμμα αποτελείται από ένα αρχείο **.F**, το οποίο περιέχει ένα κυρίως πρόγραμμα και μία υπορουτίνα. Τα σημεία του προγράμματος **RADON** που προσαρμόσθηκαν με βάση το πρόγραμμα **RRADON** προκειμένου αυτό να λειτουργήσει υπό τις συνθήκες του συνδυασμού νέου Λ/Σ και άλλης γλώσσας FORTRAN, σημαίνονται με κατάλληλη σκίαση [ ].

radon.F

```
#include "count.f.i"
      PROGRAM RADON
C     LINK BETWEEN C & F77
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.P.HARVATIS - April 88
C     FEDORA CORE 8 version by G.D.ALEXOPOULOS - April 08
      DIMENSION ICNT(3),COUNT(20,5)
      BYTE KSPACE,KDONE,M,KFLAG,YES,IY,BEL
      CHARACTER*11 FIL
      CHARACTER*17 STIME
      CHARACTER*1 CHI
      CHARACTER *20 TD
      INTEGER*4 TICS,TICS2,TSTART,TWAIT,TSAMP
      INTEGER*1 T
      ISW=0
      DATA KSPACE,KDONE,KFLAG/1H ,1H!,1H"/,YES/'Y'/,BEL/07/
      TD="/dev/ttyS0"
      PRINT 74
  74  FORMAT(/)
      CALL ITIME(ISEC,IMIN,IHRS,JD,JM,JY)
      JY1=JY-100
      WRITE(FIL,111) IHRS,IMIN,JD,JM,JY1
 111  FORMAT(3I2,'.',I2,I2)
      CALL REPLACE(FIL,'0',' ')
      PRINT 114,FIL
      OPEN(UNIT=10,FILE=FIL,STATUS='OLD',ACCESS='SEQUENTIAL',ERR=100)
      PRINT 34
  34  FORMAT(/$,'File already existing - Press Y to override ')
      READ 35,IY
  35  FORMAT(A1)
      IF(IY.NE.YES) THEN
      STOP
      ENDIF
  36  CLOSE(UNIT=10,STATUS='DELETE',ERR=102)
 100  OPEN(UNIT=10,FILE=FIL,STATUS='NEW',ACCESS='SEQUENTIAL',ERR=101)
      OPEN(11,FILE='RREXCO.DAT')
      READ(11,37)NSC,MEAS,IDT,MMEAS,PDX,TSAMP
  37  FORMAT(4I5,F10.0,I5)
      DT=IDT
      NSN=1
      WRITE(10,53)JD,JM,JY1,IHRS,IMIN,ISEC,NSC,MEAS,IDT
  53  FORMAT(I2,'-',I2,'-',I2,10X,I2,':',I2,':', I2,/3I5,2F10.3)
      ICC=0
      PAUSE ' PLS PUT DAISY CHAIN ON LINE AND PRESS FLAG'
      PRINT 1
   1  FORMAT(/'IF D/C IS ON LINE A MESSAGE FOLLOWS',
     1' IF NOT CPU LOOPS TILL D/C CONNECTED'/
     2'IF D/C IS CONNECTED BUT CPU STILL LOOPS PLEASE PRESS FLAG')
C     open RS-232 special file
      CALL OPENCNT(IDV,TD)
C     wait until the flag button is pressed
   3  CALL FMCNT(IDV,M)
      IF(M.NE.KFLAG)GO TO 3
      PRINT 4
   4  FORMAT(/'DAISY CHAIN IS ON LINE - EXPERIMENT CAN GO AHEAD'/)
      CHI='A'
      CALL TOCNT(IDV,CHI)
  40  SX=0
      SXX=0
```

```
      RN=0
      L1=MEAS
   85 DO 30 L=1,L1
C     reset counter
      CHI="R"
      CALL TOCNT(IDV,CHI)
      IF(L.GT.1)GO TO 50
      IF(ISW.EQ.0)GO TO 151
C     set RS-232 driver for immediate responce
   51 T=0
      CALL SETCNT(IDV,T)
C     read one character
   52 CALL FMCNT(IDV,M)
      IF(M.EQ.KFLAG)GO TO 45
   54 IF(M.EQ.0)GO TO 501
C     halt, reset and clear buffer
      CHI="H"
      CALL TOCNT(IDV,CHI)
      CHI="R"
      CALL TOCNT(IDV,CHI)
      CALL CLEARCNT(IDV)
C     get time since reference point
  501 CALL SECNDS(TSTART,TICS,TWAIT,TICS2)
      IF(TWAIT.LT.TSAMP)GO TO 52
C     get reference point
  151 TWAIT=TSTART
      TICS2=TICS
      TSTART=0
      TICS=0
      CALL SECNDS(TSTART,TICS,TWAIT,TICS2)
      ISW=1
      CALL ITIME(ISEC,IMIN,IHRS,JD,JM,JY)
      JY1=JY-100
      WRITE(STIME,500)JD,JM,JY1,IHRS,IMIN,ISEC
  500 FORMAT(I2,'/',I2,'/',I2,'-',I2,':',I2,':',I2)
      CALL REPLACE(STIME,'0',' ')
  141 PRINT 33,BEL,NSN
   33 FORMAT(/,A1,'Sample number : ',I4,5X,'Counting started')
C     start
   50 CHI="S"
      CALL TOCNT(IDV,CHI)
C     set RS-232 driver for normal responce (wait mode)
      T=1
      CALL SETCNT(IDV,T)
      CALL FMCNT(IDV,M)
C     led OFF
   14 CHI="B"
      CALL TOCNT(IDV,CHI)
      CALL FMCNT(IDV,M)
      IF(M.EQ.KDONE)GO TO 44
C     halt
      CHI="H"
      CALL TOCNT(IDV,CHI)
      CALL FMCNT(IDV,M)
      GO TO 44
C     check for double bush on the flag button
..45  CALL SLEEP(2)
      CALL FMCNT(IDV,M)
      IF(M.EQ.KFLAG)GO TO 41
      GO TO 54
C     get data from NSC scalers
```

```
C     led ON
  44  CHI="A"
      CALL TOCNT(IDV,CHI)
      DO 15 I=1,NSC
      CHI="I"
      CALL TOCNT(IDV,CHI)
      CALL DATAIN(IDV,IC,ICNT(I))
      CALL FMCNT(IDV,M)
      IF(M.NE.KSPACE)ICC=1
      COUNT(L,I)=REAL(ICNT(I))
      IF(IC.NE.0)ICC=1
  15  CONTINUE
      CALL TOCNT(IDV,CHI)
      CALL FMCNT(IDV,M)
      IF(COUNT(L,1).NE.DT.OR.M.NE.KDONE.OR.ICC.EQ.1)GO TO 18
      SXX=SXX+COUNT(L,2)**2
      SX=SX+COUNT(L,2)
      RN= RN+1
      GOTO 30
  18  PRINT 16,M,IC,ICC,COUNT(L,1),DT
  16  FORMAT(/'ERROR DURING PREVIOUS RUN - RUN TO BE REPEATED'
     1/' VALUES : ',' M= ',A1,'  IC=',I3,'  ICC=',I3,
     2' COUNT(1)=',F7.0,'  DT=',F7.0)
C     reset, start
      CHI="R"
      CALL TOCNT(IDV,CHI)
      CHI="S"
      CALL TOCNT(IDV,CHI)
      CALL FMCNT(IDV,M)
      GO TO 14
  30  CONTINUE
      XM=SX/RN
      PSDX=100.*SQRT((SXX-RN*XM**2)/(RN*(RN-1.)))/XM
  84  PRINT 89,NSN,IFIX(RN),IDT,(IFIX(COUNT(L,2)),L=1,MEAS),XM,PSDX
     +,STIME
  89  FORMAT(1X,I4,2X,I4,2X,I4,3(2X,I5),4X,F6.1,2X,F5.2,' % ',A17)
  87  WRITE(10,38)NSN,IFIX(RN),XM,PSDX,STIME
  38  FORMAT(I5,1X,I5,1X,F6.1,F7.2,5X,A17)
      NSN=NSN+1
      GO TO 40
C     double push on the flag button stops the program
  41  CHI="B"
      CALL TOCNT(IDV,CHI)
      CLOSE(10,STATUS='SAVE',ERR=103)
      CLOSE(11,STATUS='SAVE',ERR=103)
      CALL CLOSECNT(IDV)
      PRINT 48,BEL
  48  FORMAT(A1,/'TWO FLAGS RECEIVED - END OF THIS EXPERIMENT'//)
      PRINT 114,FIL
 114  FORMAT('Output file name : ',A12/)
      STOP
C     errors
 101  PRINT 104
 104  FORMAT(/'File cannot be opened')
      STOP
 102  PRINT 105
 105  FORMAT(/'File cannot be overriden')
      STOP
 103  PRINT 106
 106  FORMAT (/'File cannot be closed')
      STOP
```

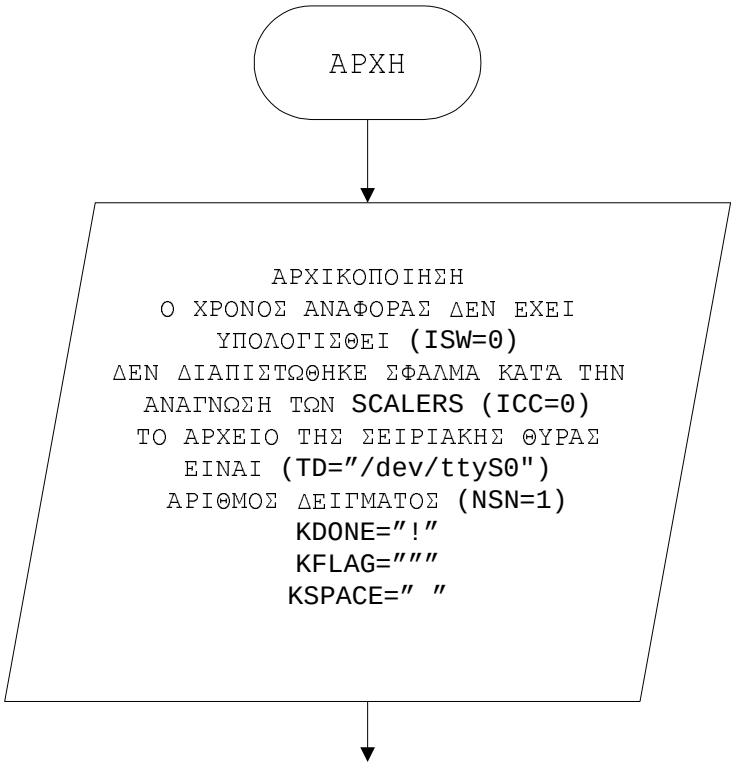
```
      END
      SUBROUTINE REPLACE(STR,C1,C2)
      CHARACTER*(*) STR
      CHARACTER C1,C2
C     replaces C2 with Cl in STR
      DO 900 I=1,LEN(STR)
      IF (STR(I:I) .EQ. C2) THEN
      STR(I:I)=C1
      ENDIF
  900 CONTINUE
      RETURN
      END
```

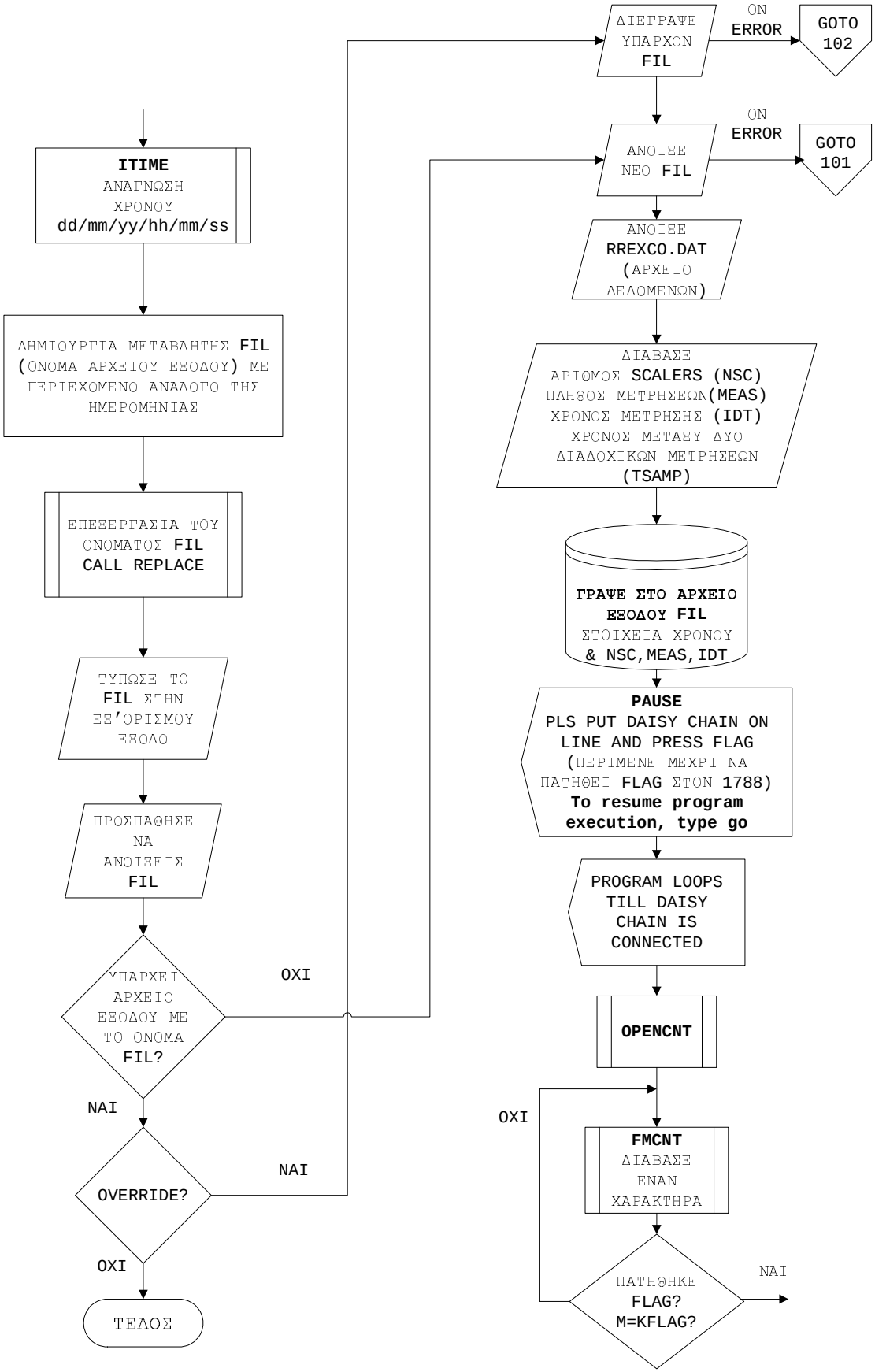
Σε σχέση με το πρόγραμμα **RRADON** (Χαρβάτης, 1990), το πρόγραμμα **RADON**, για να λειτουργήσει στα πλαίσια του Λ/Σ και της FORTRAN που χρησιμοποιείται σε αυτή τη ΔΕ, χρειάζεται τις εξής αλλαγές:

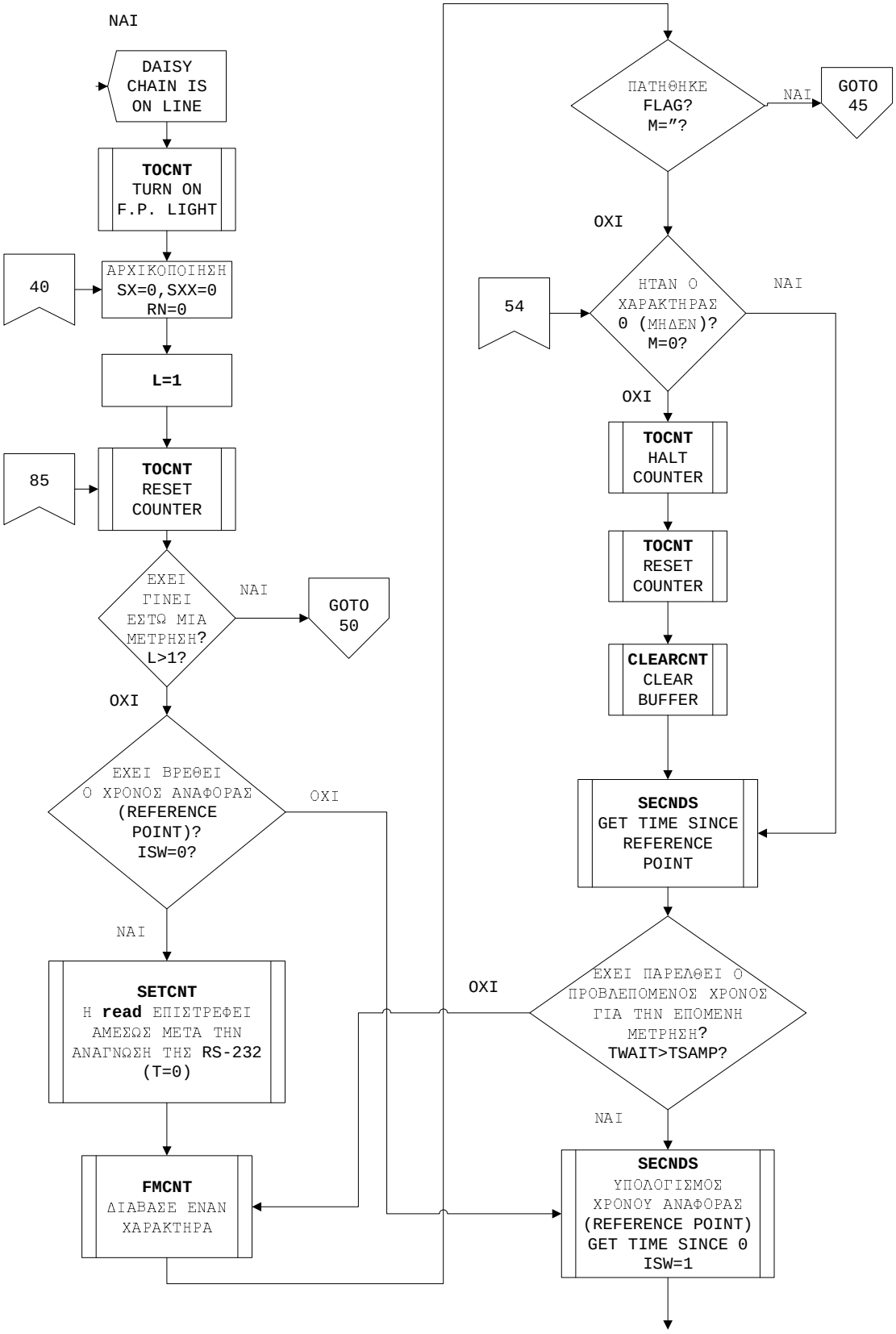
- Χρήση της κατάλληλης εντολής προεπεξεργασίας **#include**
- Αρχικοποίηση τυχόν μη αρχικοποιημένων μεταβλητών του παλαιού προγράμματος **RRADON**
- Αντικατάσταση της εντολής **ENCODE** με κατάλληλα διαμορφωμένη εντολή **WRITE**. Η εντολή **ENCODE** δεν είναι μέρος της τυποποίησης ANSI77 για την FORTRAN77. Η εντολή **WRITE** στην FORTRAN77 έχει τη δυνατότητα εγγραφής σε "εσωτερικό αρχείο" (στα αγγλικά internal file)
- Αύξηση του μήκους της μεταβλητής **FIL** (για τη σωστή ονοματοδοσία του αρχείου εξόδου)
- Εισαγωγή της μεταβλητής χαρακτήρων **CHI** και της μεταβλητής χαρακτήρων **TD** (για τη σωστή κλήση των υπορουτινών της βιβλιοθήκης **count**, **ToCnt** και **OpenCnt** αντίστοιχα)
- Εισαγωγή της μεταβλητής ακεραίων **T** (για τη σωστή κλήση της υπορουτίνας της βιβλιοθήκης **count**, **SetCnt**)
- Η μεταβλητή **TD** περιέχει το αρχείο της περιφερειακής συσκευής της σειριακής θύρας του Η/Υ που χρησιμοποιείται. Στην εδώ περίπτωση είναι το /dev/ttyS0. Στο τέλος της ακολουθίας χαρακτήρων /dev/ttyS0, δεν χρειάζεται να προστεθεί ο χαρακτήρας ASCII **NUL**, καθόσον, όπως αναφέρθηκε στο 4^{o} Κεφάλαιο, την εργασία αυτή επιτελεί η υπορουτίνα του προγράμματος **onion.F**

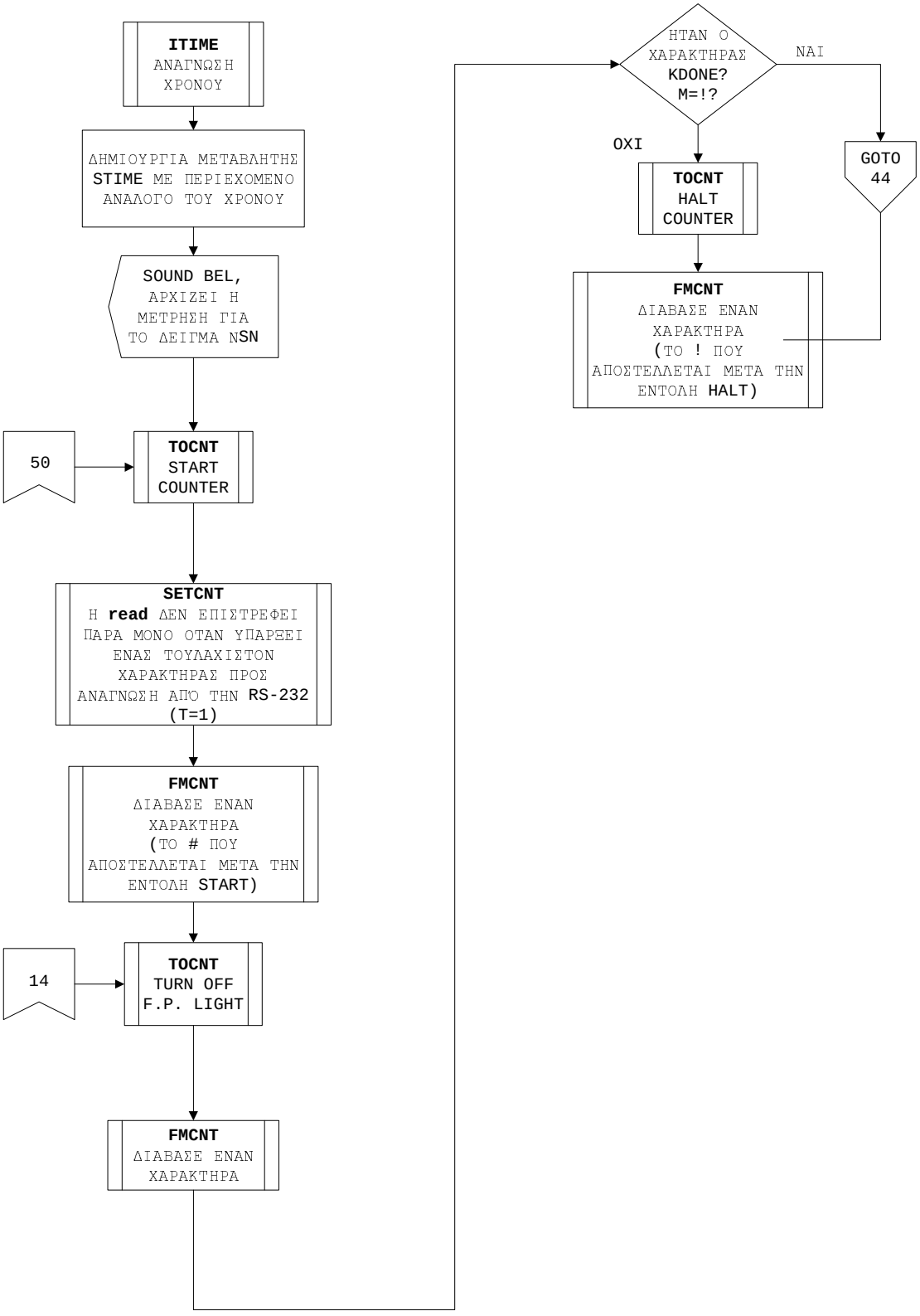
Σημειώνεται ότι τα αρχεία που παρουσιάζονται στην εδώ § 5.3.6 περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία. Το λογικό διάγραμμα του προγράμματος **RADON** για την προγραμματιστική υλοποίηση της παρούσης ΔΕ, παρουσιάζεται στη συνέχεια.

radon.F
for 1788

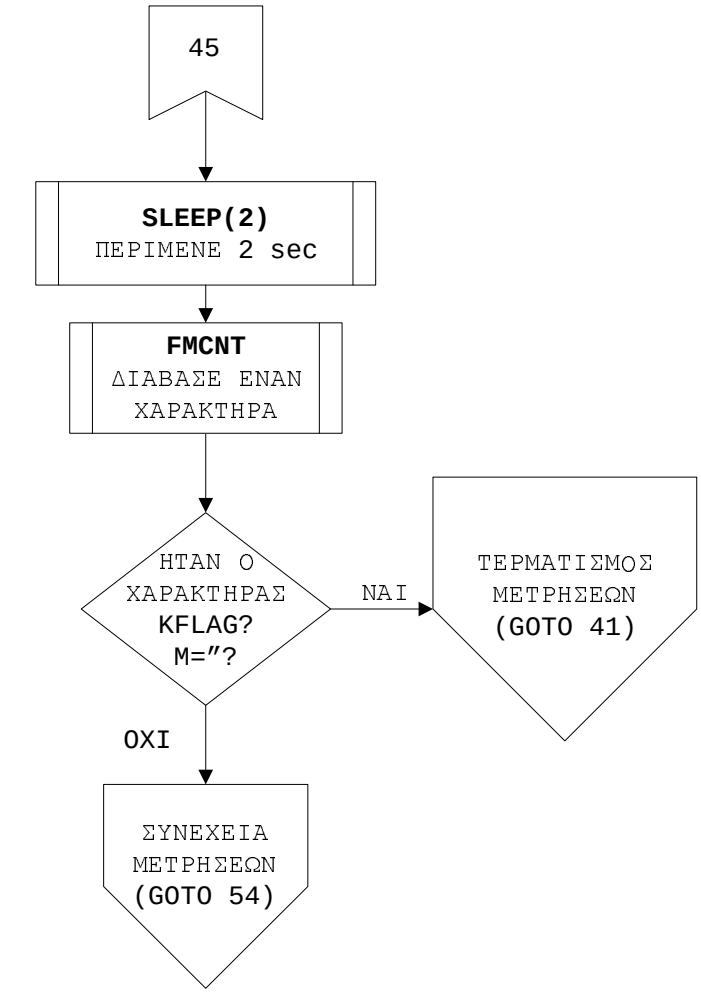



ITIME
ΑΝΑΓΝΩΣΗ
ΧΡΟΝΟΥ
dd/mm/yy/hh/mm/ss
ΔΗΜΙΟΥΡΓΙΑ ΜΕΤΑΒΛΗΤΗΣ FIL (ΟΝΟΜΑ ΑΡΧΕΙΟΥ ΕΞΟΔΟΥ) ΜΕ ΠΕΡΙΕΧΟΜΕΝΟ ΑΝΑΛΟΓΟ ΤΗΣ ΗΜΕΡΟΜΗΝΙΑΣ
ΕΠΕΞΕΡΓΑΣΙΑ ΤΟΥ ΟΝΟΜΑΤΟΣ FIL
CALL REPLACE
ΤΥΠΩΣΕ ΤΟ FIL ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
ΠΡΟΣΠΑΘΗΣΕ ΝΑ ΑΝΟΙΞΕΙΣ FIL
ΥΠΑΡΧΕΙ ΑΡΧΕΙΟ ΕΞΟΔΟΥ ΜΕ ΤΟ ΟΝΟΜΑ FIL?
ΟΧΙ
ΝΑΙ
OVERRIDE?
ΝΑΙ
ΟΧΙ
ΤΕΛΟΣ
ΔΙΕΓΡΑΨΕ ΥΠΑΡΧΟΝ FIL
ON ERROR
GOTO 102
ΑΝΟΙΞΕ ΝΕΟ FIL
ON ERROR
GOTO 101
ΑΝΟΙΞΕ RREXCO.DAT (ΑΡΧΕΙΟ ΔΕΔΟΜΕΝΩΝ)
ΔΙΑΒΑΣΕ ΑΡΙΘΜΟΣ SCALERS (NSC) ΠΛΗΘΟΣ ΜΕΤΡΗΣΕΩΝ(MEAS) ΧΡΟΝΟΣ ΜΕΤΡΗΣΗΣ (IDT) ΧΡΟΝΟΣ ΜΕΤΑΞΥ ΔΥΟ ΔΙΑΔΟΧΙΚΩΝ ΜΕΤΡΗΣΕΩΝ (TSAMP)
ΓΡΑΨΕ ΣΤΟ ΑΡΧΕΙΟ ΕΞΟΔΟΥ FIL
ΣΤΟΙΧΕΙΑ ΧΡΟΝΟΥ & NSC,MEAS,IDT
PAUSE
PLS PUT DAISY CHAIN ON LINE AND PRESS FLAG
(ΠΕΡΙΜΕΝΕ ΜΕΧΡΙ ΝΑ ΠΑΤΗΘΕΙ FLAG ΣΤΟΝ 1788)
To resume program execution, type go
PROGRAM LOOPS TILL DAISY CHAIN IS CONNECTED
OPENCNT
ΟΧΙ
FMCNT
ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ
ΠΑΤΗΘΗΚΕ FLAG? M=KFLAG?
ΝΑΙ

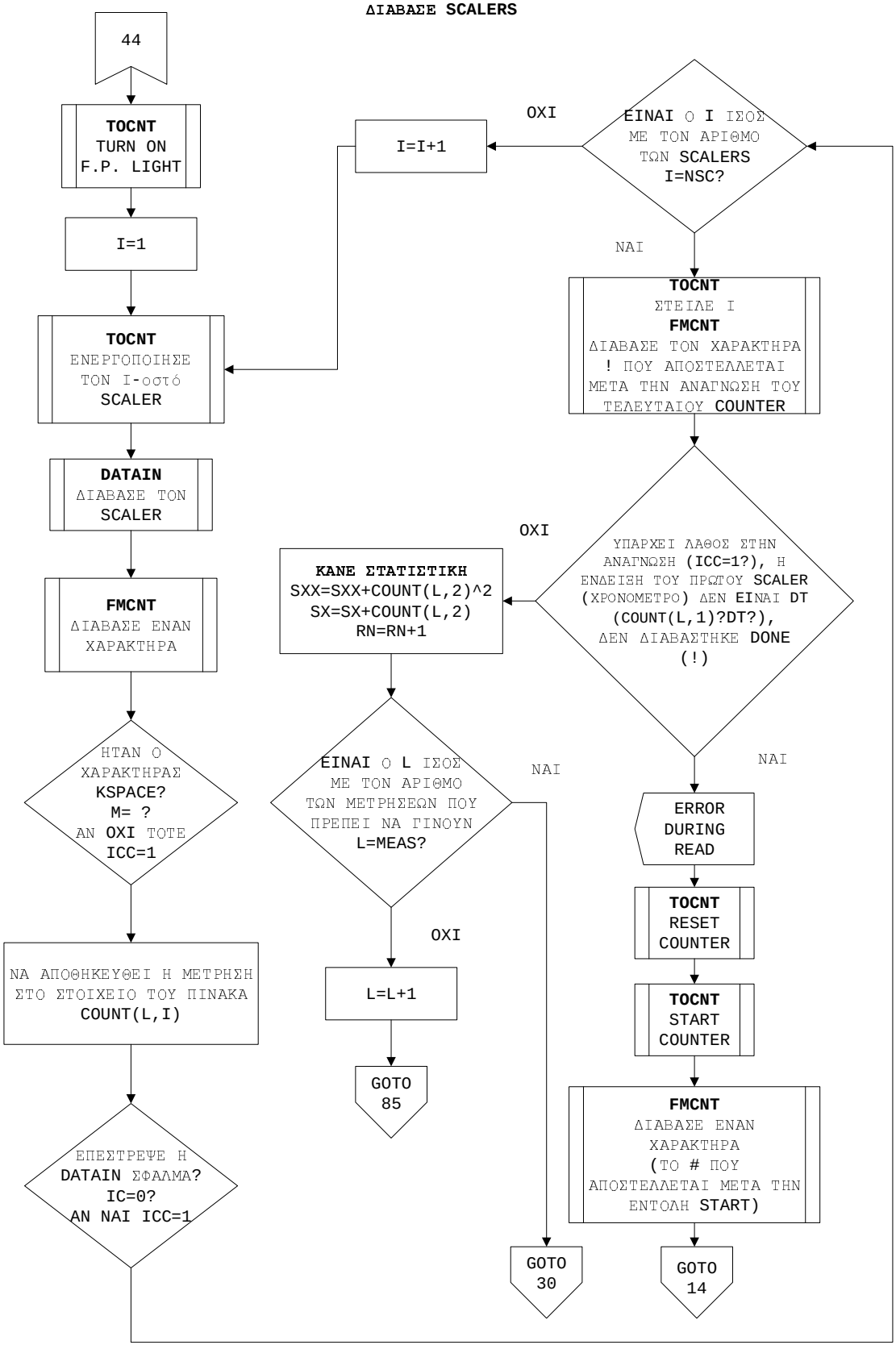

NAI
DAISY CHAIN IS ON LINE
TOCNT TURN ON F.P. LIGHT
40
ΑΡΧΙΚΟΠΟΙΗΣΗ SX=0,SXX=0 RN=0
L=1
85
TOCNT RESET COUNTER
ΕΧΕΙ ΓΙΝΕΙ ΕΣΤΩ ΜΙΑ ΜΕΤΡΗΣΗ? L>1?
NAI
GOTO 50
OXI
ΕΧΕΙ ΒΡΕΘΕΙ Ο ΧΡΟΝΟΣ ΑΝΑΦΟΡΑΣ (REFERENCE POINT)? ISW=0?
OXI
NAI
SETCNT Η read ΕΠΙΣΤΡΕΦΕΙ ΑΜΕΣΩΣ ΜΕΤΑ ΤΗΝ ΑΝΑΓΝΩΣΗ ΤΗΣ RS-232 (T=0)
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ
ΠΑΤΗΘΗΚΕ FLAG? M=″?
NAI
GOTO 45
OXI
54
ΗΤΑΝ Ο ΧΑΡΑΚΤΗΡΑΣ 0 (ΜΗΔΕΝ)? M=0?
NAI
OXI
TOCNT HALT COUNTER
TOCNT RESET COUNTER
CLEARCNT CLEAR BUFFER
SECNDS GET TIME SINCE REFERENCE POINT
ΕΧΕΙ ΠΑΡΕΛΘΕΙ Ο ΠΡΟΒΛΕΠΟΜΕΝΟΣ ΧΡΟΝΟΣ ΓΙΑ ΤΗΝ ΕΠΟΜΕΝΗ ΜΕΤΡΗΣΗ? TWAIT>TSAMP?
OXI
NAI
SECNDS ΥΠΟΛΟΓΙΣΜΟΣ ΧΡΟΝΟΥ ΑΝΑΦΟΡΑΣ (REFERENCE POINT) GET TIME SINCE 0 ISW=1

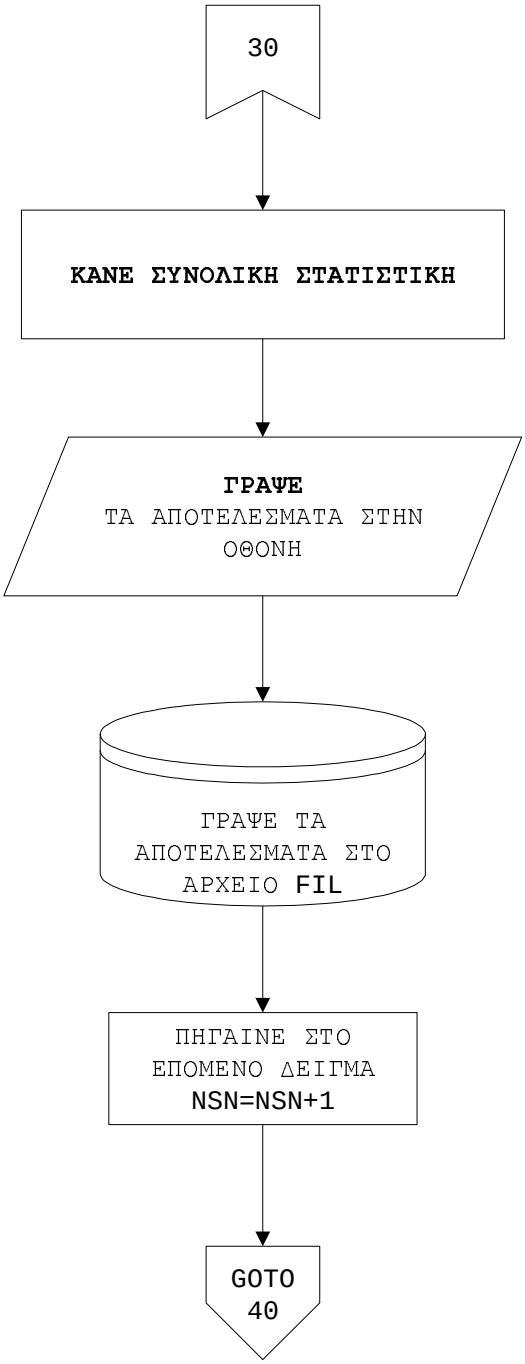

ITIME
ΑΝΑΓΝΩΣΗ ΧΡΟΝΟΥ
ΔΗΜΙΟΥΡΓΙΑ ΜΕΤΑΒΛΗΤΗΣ STIME ΜΕ ΠΕΡΙΕΧΟΜΕΝΟ ΑΝΑΛΟΓΟ ΤΟΥ ΧΡΟΝΟΥ
SOUND BEL, ΑΡΧΙΖΕΙ Η ΜΕΤΡΗΣΗ ΓΙΑ ΤΟ ΔΕΙΓΜΑ NSN
50
TOCNT START COUNTER
SETCNT
Η read ΔΕΝ ΕΠΙΣΤΡΕΦΕΙ ΠΑΡΑ ΜΟΝΟ ΟΤΑΝ ΥΠΑΡΞΕΙ ΕΝΑΣ ΤΟΥΛΑΧΙΣΤΟΝ ΧΑΡΑΚΤΗΡΑΣ ΠΡΟΣ ΑΝΑΓΝΩΣΗ ΑΠΟ ΤΗΝ RS-232 (T=1)
FMCNT
ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΤΟ # ΠΟΥ ΑΠΟΣΤΕΛΛΕΤΑΙ ΜΕΤΑ ΤΗΝ ΕΝΤΟΛΗ START)
14
TOCNT TURN OFF F.P. LIGHT
FMCNT
ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ
ΗΤΑΝ Ο ΧΑΡΑΚΤΗΡΑΣ KDONE? M=!?
ΝΑΙ
ΟΧΙ
GOTO 44
TOCNT HALT COUNTER
FMCNT
ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΤΟ ! ΠΟΥ ΑΠΟΣΤΕΛΛΕΤΑΙ ΜΕΤΑ ΤΗΝ ΕΝΤΟΛΗ HALT)

ΑΝ ΠΑΤΗΘΕΙ 2° FLAG ΝΑ ΤΕΡΜΑΤΙΣΘΟΥΝ ΟΙ ΜΕΤΡΗΣΕΙΣ

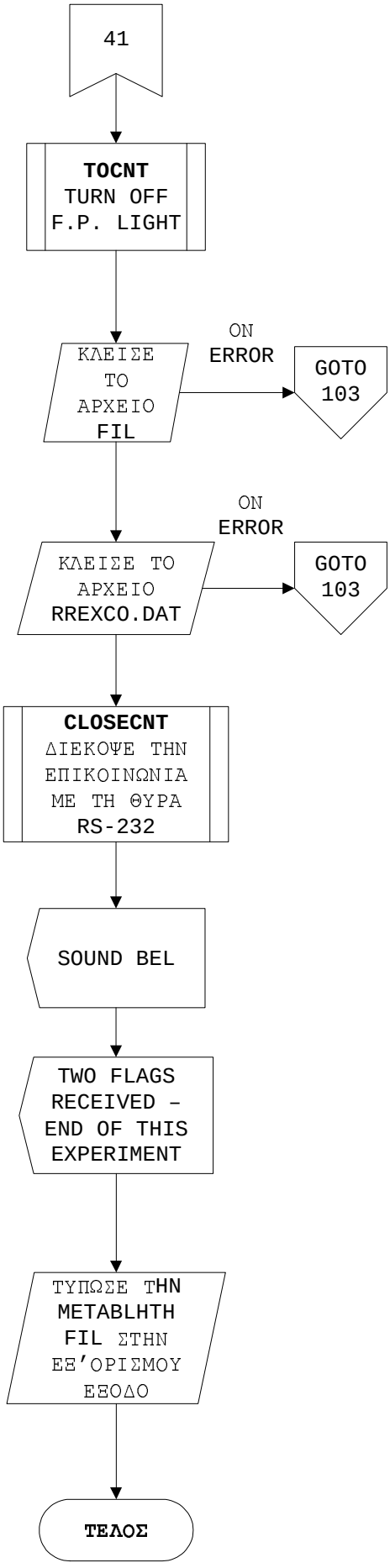


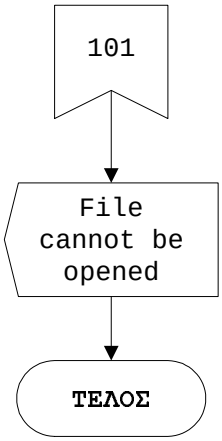
ΔΙΑΒΑΣΕ SCALERS

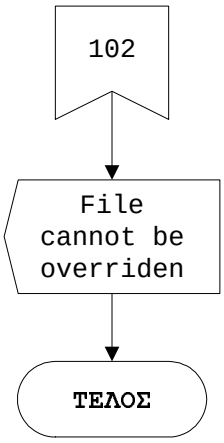



30
ΚΑΝΕ ΣΥΝΟΛΙΚΗ ΣΤΑΤΙΣΤΙΚΗ
ΓΡΑΨΕ
ΤΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΣΤΗΝ
ΟΘΟΝΗ
ΓΡΑΨΕ ΤΑ
ΑΠΟΤΕΛΕΣΜΑΤΑ ΣΤΟ
ΑΡΧΕΙΟ FIL
ΠΗΓΑΙΝΕ ΣΤΟ
ΕΠΟΜΕΝΟ ΔΕΙΓΜΑ
NSN=NSN+1
GOTO
40

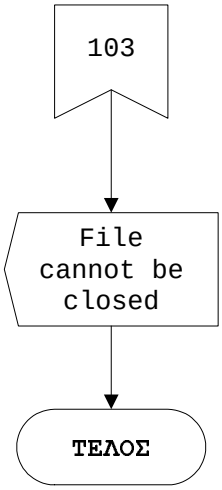
ΤΕΡΜΑΤΙΣΜΟΣ ΜΕΤΡΗΣΕΩΝ





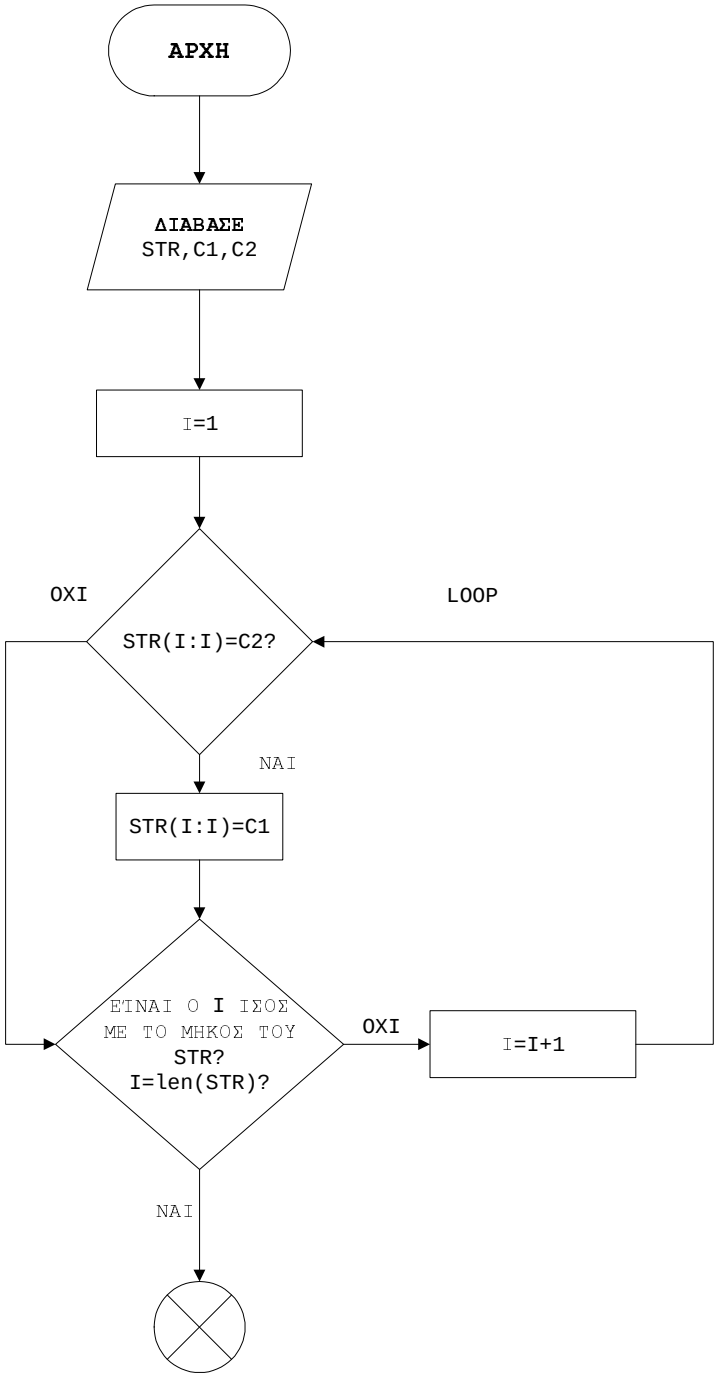


ΣΦΑΛΜΑΤΑ ΜΕ ΑΡΧΕΙΑ



SUBROUTINE REPLACE
FOR
radon.F

ΑΝ Η ΑΚΟΛΟΥΘΙΑ ΧΑΡΑΚΤΗΡΩΝ
STR
ΕΧΕΙ ΚΕΝΑ (C2)
ΤΟΤΕ ΑΝΤΙΚΑΤΕΣΤΗΣΕ ΤΑ ΜΕ
ΜΗΔΕΝ (C1)



5.3.7 Μετάφραση και σύνδεση

Λαμβάνοντας υπόψιν τα όσα σημειώθηκαν στη παράγραφο 4.8 και δεδομένου ότι το πρόγραμμα **RADON** χρησιμοποιεί για να λειτουργήσει τις υπορουτίνες της βιβλιοθήκης **count**, η μετάφραση και η σύνδεσή του για την παραγωγή του αντίστοιχου εκτελέσιμου **radon** θα μπορούσε να γίνει από την εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –g *radon*.F –l count –o *radon*

η οποία μπορεί να αποτελεί και μέρος άλλου ευρύτερου makefile.

5.4 Μονάδα επικοινωνίας 2088

5.4.1 Γενικά

Η μονάδα επικοινωνίας CANBERRA 2088 που θα χρησιμοποιηθεί έχει σειριακό αριθμό 987530 και διακρίνεται στις φωτογραφίες των σχημάτων 2.1 (εμπρόσθια όψη) και 2.2 (οπίσθια όψη) του αντίστοιχου εγχειριδίου (βλ. και Παράρτημα). Στη συνέχεια παρουσιάζονται τα εξωτερικά λειτουργικά μέρη της μονάδας που είναι απαραίτητα για την παρούσα ΔΕ. Τα λοιπά λειτουργικά μέρη αγνοούνται. Έτσι λοιπόν:

Στο εμπρόσθιο μέρος υπάρχουν

- διακόπτης ON/OFF. Για να λειτουργεί η μονάδα κατά τον τρόπο που είναι επιθυμητός, ο διακόπτης αυτός πρέπει να είναι στη θέση ON
- διακόπτης FORMAT LINE/CONTINUOUS. Για να λειτουργεί η μονάδα κατά τον τρόπο που είναι επιθυμητός, ο διακόπτης αυτός πρέπει να είναι στη θέση FORMAT LINE
- διακόπτης SINGLE/RECYCLE. Για να λειτουργεί η μονάδα κατά τον τρόπο που είναι επιθυμητός, ο διακόπτης αυτός πρέπει να είναι στη θέση SINGLE

Στο πίσω μέρος υπάρχουν

- μία θύρα εισόδου 15 pin (CONTROL IN) η οποία συνδέει σε σειρά (Daisy Chain) την μονάδα με αντίστοιχο κατάλληλο καταμετρητή.
- μία θύρα εξόδου 15 pin (CONTROL OUT) η οποία συνδέει σε σειρά (Daisy Chain) την μονάδα με αντίστοιχο κατάλληλο καταμετρητή.

- μία θύρα (EIA DATA) επικοινωνιών τύπου RS-232 female 25 pin (η μονάδα χρησιμοποιείται ως εξοπλισμός DTE).

Η υπόψιν μονάδα αγοράσθηκε για τους σκοπούς της παρούσης ΔΕ, προκειμένου να μην απασχοληθεί καθόλου η αντίστοιχη μονάδα, η οποία βρίσκεται σε ενεργή λειτουργία στο ΕΠΤ-ΕΜΠ προκειμένου να πραγματοποιούνται συνεχώς "Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI"

5.4.2 Ρύθμιση

Υπενθυμίζεται εδώ ότι, κατά την κατάστρωση του λογισμικού, οι υπορουτίνες ρυθμίσεως της σειριακής θύρας RS-232 του Η/Υ, γράφθηκαν ώστε να τηρούνται οι ακόλουθες προδιαγραφές:

Ταχύτητα Μετάδοσης 9600 baud.

Μήκος χαρακτήρα 7 bits.

Parity EVEN.

Ένα stop bit

Όπως πληροφορεί το εγχειρίδιο της συσκευής, αυτή όπως έρχεται προρυθμισμένη από το εργοστάσιο κατασκευής έχει τους εσωτερικούς "βραχυκυκλωτήρες" (στα αγγλικά jumbers) ρύθμισης στις θέσεις:

1) Βραχυκυκλωτήρας A, non A στη θέση non A
2) Βραχυκυκλωτήρας B, non B στη θέση non B
3) Βραχυκυκλωτήρας 6C, 8C στη θέση 6C (σημαίνει "6 χαρακτήρες για κάθε λέξη δεδομένων)
4) Βραχυκυκλωτήρας 10L, 8L στη θέση 10L (σημαίνει "10 λέξεις ανά γραμμή")
5) Βραχυκυκλωτήρας C3, non C3 στη θέση C3 (σημαίνει ότι τα pins 5, 8, 11, 20 της σειριακής θύρας RS-232 του 2088, είναι ενεργοποιημένα για hardware handshaking, τα pins αυτά δεν χρησιμοποιούνται για την εδώ σειριακή επικοινωνία)
6) Βραχυκυκλωτήρας XL, non XL στη θέση XL (σημαίνει ότι παράγονται 5V στην ομοαξονική έξοδο BNC SIG.OUT)
7) Βραχυκυκλωτήρας X, non X στη θέση X (σημαίνει ότι τα πιοπάνω 5V είναι +5V)
8) Βραχυκυκλωτήρας M1, C1 στη θέση C1

9) Βραχυκυκλωτήρας M2, C2 στη θέση C2

10) Βραχυκυκλωτήρας NDLY, DLY στη θέση NDLY

11) Βραχυκυκλωτήρας ET, non ET στη θέση ET

12) Το ολοκληρωμένο 370051 στη θέση A2

13) Το ολοκληρωμένο 370052 στη θέση A23

14) Οι μικροδιακόπτες (dip switches) S1 – S8 στις θέσεις 1:OFF, 2:ON, 3:OFF, 4:OFF, 5:ON, 6:OFF, 7:ON, 8:OFF

Προφανώς, πριν την σύνδεση της συσκευής έγινε ο απαραίτητος έλεγχος των βραχυκυκλωτήρων ώστε αυτή να είναι συμβατή με τις ρυθμίσεις της σειριακής θύρας RS-232 του Η/Υ, αλλά και με την ήδη υπάρχουσα στο ΕΠΤ-ΕΜΠ μονάδα επικοινωνίας 2088. Για το σκοπό αυτό εξετάσθηκαν οι ρυθμίσεις των "βραχυκυκλωτήρων" που ήταν σημειωμένες στο αντίστοιχο εγχειρίδιο, ελέγχθηκαν οι ορθές θέσεις και ρυθμίσθηκαν τα ακόλουθα:

1) Βραχυκυκλωτήρας A, non A στη θέση A (δίνει -12 V στο pin 10 της σειριακής θύρας RS-232 του 2088, το pin αυτό δεν χρησιμοποιείται για την εδώ σειριακή επικοινωνία)

2) Βραχυκυκλωτήρας B, non B στη θέση B (δίνει +12 V στο pin 9 της σειριακής θύρας RS-232 του 2088, το pin αυτό δεν χρησιμοποιείται για την εδώ σειριακή επικοινωνία)

3) Βραχυκυκλωτήρας M1, C1 στη θέση M1 (σημαίνει ότι το pin 2 της σειριακής θύρας RS-232 του 2088, είναι "transmitted data" – TxD)

4) Βραχυκυκλωτήρας M2, C2 στη θέση M2 (σημαίνει ότι το pin 3 της σειριακής θύρας RS-232 του 2088, είναι "received data" – RxD)

5) Βραχυκυκλωτήρας NDLY, DLY στη θέση DLY (σημαίνει ότι μετά από κάθε carriage return υπάρχει καθυστέρηση 160 msec)

6) Βραχυκυκλωτήρας ET, non ET στη θέση non ET (σημαίνει ότι ο printer termination character ή αλλιώς το output termination character είναι το SUB, αν το ολοκληρωμένο 370052 είναι στη θέση A2 και το ολοκληρωμένο 370051 είναι στη θέση A23)

7) Το ολοκληρωμένο 370051 στη θέση A23 (θέση κατάλληλη για σύνδεση με Η/Υ)

8) Το ολοκληρωμένο 370052 στη θέση A2 (θέση κατάλληλη για σύνδεση με Η/Υ)

9) Οι μικροδιακόπτες (dip switches) S1 – S8 στις θέσεις 1:OFF, 2:ON, 3:ON, 4:ON, 5:ON, 6:ON, 7:ON, 8:ON (σημαίνει ότι η σειριακή θύρα ρυθμίζεται ως EVEN PARITY, 7 BITS, 1 STOP BIT, PARITY ENABLED, 9600 BAUD)

Η μονάδα επικοινωνίας 2088, με τις πιοπάνω ρυθμίσεις μπορεί να χρησιμοποιείται στο ΕΠΤ-ΕΜΠ για την εφαρμογή " Μετρήσεις συγκέντρωσης ^{137}Cs ή/και θυγατρικών ^{222}Rn στην ατμόσφαιρα με ανιχνευτή NaI " στην οποία όπως έχει καταγραφεί και στο 2^{o} Κεφάλαιο, συνδέεται με την σειριακή θύρα ενός Η/Υ HP-9000/320 με λειτουργικό σύστημα UNIX. Σημειώνεται ότι η μονάδα επικοινωνίας 2088 είναι κατασκευασμένη έτσι ώστε να "διαβάζει" ψηφιακούς καταχωρητές από αντίστοιχους κατάλληλους καταμετρητές μήκους έξι χαρακτήρων.

5.4.3 Πρωτόκολλο επικοινωνίας

Η μονάδα επικοινωνίας διαθέτει πρωτόκολλο σειριακής σύνδεσης μέσω του οποίου η σειριακή θύρα του συνδεδεμένου Η/Υ είναι δυνατόν να ελέγχει έναν ή περισσότερους καταμετρητές και να παραλαμβάνει δεδομένα από αυτούς. Το πρωτόκολλο αυτό έχει δύο συνιστώσες, το πρωτόκολλο αποστολής και το πρωτόκολλο λήψης, ως εξής: (α) Αποστολή χαρακτήρων ASCII ≡ Εντολές ελέγχου από τον Η/Υ προς τη μονάδα επικοινωνίας 2088, και (β) Λήψη χαρακτήρων ASCII ≡ Πληροφορίες από την μονάδα επικοινωνίας προς τον Η/Υ. Τα δύο πρωτόκολλα είναι (όπως δίνονται από το αντίστοιχο εγχειρίδιο, βλ. και Παράρτημα):

Πρωτόκολλο αποστολής από τον Η/Υ

ASCII Code (Hex)	**Χαρακτήρας**	**Ορισμός**
52	R	(R key)clears data
53	S	(S key) starts counting
18	CAN	(CTL-X key) stops count and causes readout of data from modules. Readout also begins when any module reaches preset
58	X	(X key) Output pulse on rear panel BNC (SIG. OUT)

11	DC1	(CTL-Q ή XON) software handshaking
13	DC3	(CTL-S ή XOFF) software hanshaking

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που δέχεται η μονάδα 2088 από τον Η/Υ ερμηνεύονται ως εντολές που αναλύονται ως εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
82	R	όταν ο Η/Υ αποστέλλει ένα "R" η μονάδα επικοινωνίας 2088 αποστέλλει ένα σήμα RESET σε όλους τους καταμετρητές που είναι συνδεδεμένοι στην σειρά Daisy Chain, με αποτέλεσμα όλοι οι scalers να μηδενίσουν. Ο χαρακτήρας "R" δεν επηρεάζει τη μονάδα επικοινωνίας 2088.
83	S	όταν ο Η/Υ αποστέλλει ένα "S" η μονάδα επικοινωνίας 2088 αποστέλλει ένα σήμα START σε όλους τους καταμετρητές που είναι συνδεδεμένοι στην σειρά Daisy Chain, με αποτέλεσμα όλοι οι scalers να ξεκινούν καταγραφή. Ο χαρακτήρας "S" δεν επηρεάζει τη μονάδα επικοινωνίας 2088
24	CAN	όταν ο Η/Υ αποστέλλει ένα CAN η μονάδα επικοινωνίας 2088 αποστέλλει ένα σήμα STOP σε όλους τους καταμετρητές που είναι συνδεδεμένοι στην σειρά Daisy Chain, με αποτέλεσμα όλοι οι scalers να σταματήσουν την καταγραφή. Στη συνέχεια η μονάδα 2088 αποστέλλει στους καταμετρητές της σειράς Daisy Chain μία εντολή

		HOLD[2], διαβάζει τους scalers και μεταδίδει τα δεδομένα προς τον Η/Υ. Ειδικά για την περίπτωση που έχει επιλεγεί RECYCLE στη μονάδα 2088, τότε μετά την ανάγνωση και την αποστολή των δεδομένων, η μονάδα 2088 αποστέλλει προς τους καταμετρητές την ακολουθία εντολών RESET + START
88	X	όταν ο Η/Υ αποστέλλει ένα "X" η μονάδα επικοινωνίας 2088 απαντάει με έναν θετικό ή έναν αρνητικό (επιλέγεται από τους κατάλληλους εσωτερικούς βραχυκυκλωτήρες XL, non XL και X, non X) παλμό τάσεως διαρκείας 4 μs στην ομοαξονική έξοδο BNC SIG.OUT
19	DC3	όταν ο Η/Υ αποστέλλει ένα "XOFF" η μονάδα επικοινωνίας 2088 σταματάει να αποστέλλει δεδομένα προς τον Η/Υ. Αν το XOFF μεταδοθεί κατά τη διάρκεια αποστολής δεδομένων, η αποστολή σταματάει αφού συμπληρωθεί η λήψη του χαρακτήρα που μεταδίδεται εκείνη τη στιγμή. Σημειώνεται ότι το XOFF απλώς σταματά αλλά δεν ακυρώνει την αποστολή δεδομένων, δηλαδή τα δεδομένα, αν δεν έχουν εξαντληθεί περιμένουν για να μεταδοθούν. Όσο δεν μεταδίδονται η μονάδα παραμένει όπως λέγεται "κρεμασμένη".

[2] Για τις εσωτερικές εντολές Daisy Chain, δες (α) Instuction Manual 1788 (1978), σελ.

17	DC1	όταν ο Η/Υ αποστέλλει ένα "XON" επιτρέπεται στη μονάδα επικοινωνίας 2088 να αρχίσει ή να συνεχίσει την αποστολή δεδομένων προς τον Η/Υ.

Πρωτόκολλο λήψης προς τον Η/Υ

ASCII Code (octal)	**Χαρακτήρας**	**Ορισμός**
015	CR	Follows every transmitted word
032	SUB	Follows last word from modules
060 έως 067	0 έως 7	Data from modules
070, 071	8, 9	Data from modules

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που αποστέλλει η μονάδα 2088 στον τον Η/Υ είναι πληροφορίες που σημαίνουν τα εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
13	CR	Τερματικός χαρακτήρας μίας εξαψήφιας ένδειξης
26	SUB	Τερματικός χαρακτήρας όλων των ενδείξεων
48 έως 57	0 έως 9	Ψηφία που κατά περίπτωση συγκροτούν το περιεχόμενο μιας εξαψήφιας ένδειξης

Είναι πολύ σημαντικό να παρατηρηθεί εδώ ότι η μονάδα επικοινωνίας 2088, μπορεί να έχει ως μονάδα μέτρησης δεδομένων τόσο τη λεγόμενη "λέξη" (των έξι χαρακτήρων), όσο και τον μοναδικό χαρακτήρα, ενώ η προηγούμενή της που εξετάσθηκε, δηλ. η 1788, έχει ως μονάδα μέτρησης δεδομένων μόνο τον χαρακτήρα. Στην μονάδα επικοινωνίας 2088, λέξη από λέξη διαχωρίζεται από τον χαρακτήρα CR. Αυτά πρακτικά σημαίνουν ότι η ανάγνωση των λέξεων που μεταδίδονται προς τον Η/Υ από τη μονάδα 2088 δια μέσου της σειριακής θύρας RS-232, μπορεί να γίνεται είτε χαρακτήρα – χαρακτήρα σε non-canonical mode, είτε λέξη – λέξη σε canonical mode.

3-2 και (β) Operator's Manual 2088 (1987), σελ. 5 και 13

Σημειώνεται επίσης ότι η μονάδα 2088 μπορεί να χρησιμοποιήσει το πρωτόκολλο XON – XOFF για επικοινωνία με την σειριακή θύρα ενός Η/Υ. Οι χαρακτήρες αυτοί στην εδώ περίπτωση μπορούν να σταλούν μόνο από τον Η/Υ προς τη σειριακή θύρα της μονάδας 2088. Επισημαίνεται βέβαια ότι στις υπορουτίνες ρύθμισης της σειριακής θύρας RS-232 του Η/Υ, που συμπεριλαμβάνονται στην βιβλιοθήκη count, το πρωτόκολλο XON – XOFF είναι απενεργοποιημένο.

Στη συνέχεια δίνεται μια τυπική ακολουθία χαρακτήρων, όπως παράγεται από την επικοινωνία Η/Υ – μονάδας 2088 – z καταμετρητών 1776 σε σύνδεση Daisy Chain (τα κόμματα δεν μετράνε στην ακολουθία):

SP,SP
$1_a1_a1_a1_a1_a1_a$,CR,$2_a2_a2_a2_a2_a2_a$,CR,
SP,SP
$1_b1_b1_b1_b1_b1_b$,CR,$2_b2_b2_b2_b2_b2_b$,CR,
…….
SP,SP
$1_z1_z1_z1_z1_z1_z$,CR,$2_z2_z2_z2_z2_z2_z$,CR,SUB

όπου

a ο πρώτος καταμετρητής

b ο δεύτερος καταμετρητής

z ο τελευταίος καταμετρητής

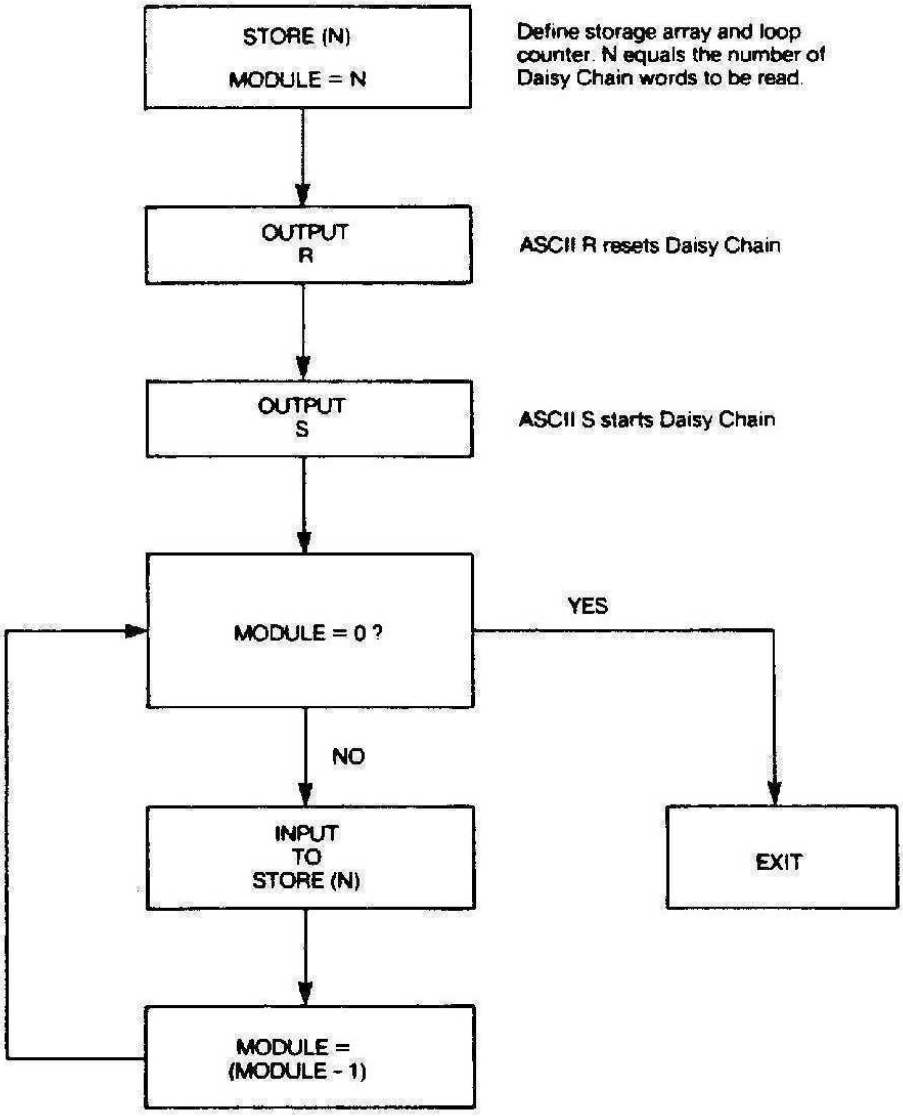
SP,SP Space,Space δηλ. δύο κενά που δηλώνουν την μετάβαση της ανάγνωσης από καταμετρητή σε καταμετρητή

CR Carriage Return δηλ. ο χαρακτήρας που δηλώνει το τέλος των δεδομένων (έξι χαρακτήρες) ενός scaler έξι χαρακτήρων

και

SUB χαρακτήρας που δηλώνει το τέλος των δεδομένων από όλους τους καταμετρητές

Έτσι, σύμφωνα με το εγχειρίδιο της μονάδας 2088 (βλ. και Παράρτημα), και την πιοπάνω ανάλυση, μια τυπική συνδιάλεξη μεταξύ Η/Υ και 2088 μπορεί να έχει μορφή σύμφωνα με το ακόλουθο λογικό διάγραμμα:



Σύμφωνα με αυτό το λογικό διάγραμμα στην έναρξη της διαδικασίας το αντίστοιχο πρόγραμμα περιμένει να τελειώσει η τρέχουσα μέτρηση και να αποσταλούν τα δεδομένα προς τη σειριακή θύρα του Η/Υ. Η μέτρηση μπορεί να τελειώσει αν εξαντληθεί ο προρυθμισμένος χρόνος μέτρησης στους καταμετρητές ή αν πατηθεί το MANUAL PRINT button,ή αν πατηθεί οποιοδήποτε STOP button σε οποιοδήποτε από τους συνδεδεμένους καταμετρητές, ή αν ληφθεί CAN από τον Η/Υ προς την μονάδα 2088. Για την υλοποίηση του υπόψιν λογικού διαγράμματος υποτίθεται ότι ο Η/Υ είναι ικανός να δέχεται δεδομένα με αρκετή ταχύτητα, χωρίς να χάνει χαρακτήρες.

Σύμφωνα με το εγχειρίδιο της μονάδας 2088, το πιοπάνω λογικό διάγραμμα θα μπορούσε να υλοποιηθεί με το ακόλουθο πρόγραμμα σε FORTRAN IV για Η/Υ VAX 11/750 ή Η/Υ PDP 11:

```
ccccc
c     Test program to demonstrate a simple application for the
c     Model 2088 with the End of Transmission character
c     SUB (1A HEX) used to signal the computer that the last
c     Daisy Chain Word has been read.
c
c     The Model 2088 is set for SINGLE cycle and is configured
c     with the Computer Readout Format PROM and the ET-ET/ jumper
c     in the ET/ position.
c
c     The computer resets and starts the Daisy Chain, and readout
c     is started when a counting module in the Daisy Chain reaches
c     preset or is manually stopped.
c
c     This program was tested with a VAX*11/750 and a PDP*11.
c
c     Lun 10 was externally assigned to the terminal port connected
c     to the Model 2088 EIA output.
c
c
      integer     lun, nmods, res, sta
      integer*2   nimdat (30)          !maximum of 30 NIM data
                                       !words chosen
      lun = 10                         !establish 2088 line
      res = 82                         !R command ascii code
      sta = 83                         !S command ascii code
10    write (lun, 11) res, sta         !reset 2088; start counting
11    format ('+', 2a1, $)
      i = 1                            !start reading data back;
15    read (lun, 20, end = 35) nimdat (i)  !record returned count,
                                       !end if SUB received
      i = i + 1
      goto 15
20    format (i8)
35    rewind lun                       !move file pointer to beg.
      nmods = i-1
      type 39 nmods
39    format (' ',i4,' words read back: ', $)  !read out header
      do 50 i = 1, nmods               !     and
      write (5, 40) nimdat(i)          !module data
40    format ('+ ', i6, $)
50    continue
      goto 10
      end
c
c
c     End of example.
c
ccccc
```

*VAX and PDP are registered trademarks of Digital Equipment Corporation

5.4.4 Σύνδεση με την πειραματική διάταξη

Μετά τη ρύθμιση η υπόψιν μονάδα επικοινωνίας τοποθετήθηκε στην μονάδα NIM Bin μαζί με τον καταμετρητή CANBERRA – Model 1776. Ο καταμετρητής συνδέθηκε με τη μονάδα επικοινωνίας με καλωδίωση Daisy Chain, με τον τρόπο που διακρίνεται στην φωτογραφία του Διαγράμματος 5.11. Η μονάδα επικοινωνίας συνδέθηκε με το αρσενικό βύσμα DB25 σε αναμονή, το οποίο διακρίνεται στη φωτογραφία του Διαγράμματος 5.6.

5.4.5 Προγραμματιστική υλοποίηση από τον Σιμόπουλο (1994)

Ένα ολοκληρωμένο παράδειγμα χρησιμοποίησης των υπορουτινών της βιβλιοθήκης **count,** όπως αυτή αναλύεται στο κείμενο της ΔΕ του Χαρβάτη (1990), για συνεργασία με μονάδα επικοινωνίας 2088, προκύπτει από διαδοχικές μετασκευές του προγράμματος FORTRAN IV της προηγούμενης παραγράφου, ώστε να λειτουργεί στο Λ/Σ HP-UX 6.x και 7.x.. Το πρόγραμμα αυτό είναι γνωστό στο ΕΠΤ-ΕΜΠ ως **CONTRL** και δηλώνεται ως κατάλληλο για τον έλεγχο τόσο της μονάδας επικοινωνίας 1788, όσο και για τον έλεγχο της μονάδας επικοινωνίας 2088. Τα σημεία του προγράμματος που πρέπει να αλλάξουν, προκειμένου αυτό να προσαρμοσθεί για περιβάλλον Λ/Σ LINUX / POSIX και γλώσσα προγραμματισμού FORTRAN77, σημαίνονται με κατάλληλη σκίαση [ ]. Τονίζεται ότι για την ορθή λειτουργία του προγράμματος **CONTRL**, η βιβλιοθήκη **count**, όπως είχε δημιουργηθεί από τον Χαρβάτη (1990) τροποποιήθηκε ώστε να συμπεριλάβει εκδόσεις των υπορουτινών **opencnt** και **setcnt**, οι οποίες να υποστηρίζουν canonical mode.

Περιγραφή: Στο ΕΠΤ-ΕΜΠ, η μονάδα 2088 μπορεί να συνδέεται συνήθως με έναν καταμετρητή/χρονόμετρο (counter/timer) CANBERRA - Model 1776 και έναν ή περισσότερους καταμετρητές (counters), κατά προτίμηση επίσης CANBERRA – Model 1776, χονδρικά κατά τη φωτογραφία του Διαγράμματος 2.14. Η διάταξη αυτή, που ονομάζεται διάταξη counter-timer, λειτουργεί όπως περιγράφθηκε ήδη στην παράγραφο 5.3.5. Το πρόγραμμα **CONTRL** χρησιμοποιείται για τον έλεγχο της καλής λειτουργίας των διατάξεων counter-timer, ανάλογα με τις εντολές που αποστέλλονται στις μονάδες επικοινωνίας 1788 ή 2088 και για την προβολή σε τερματικό των μετρήσεών τους.

Οδηγίες Χρήσεως: Το πρόγραμμα **CONTRL** τρέχει με την απλή εντολή :

contrl

Στην οθόνη εμφανίζεται το μήνυμα :

GIVE COMMAND CHARACTER

Ο χρήστης καλείται να δώσει την κατάλληλη εντολή - χαρακτήρα[3] προς τη μονάδα επικοινωνίας ή την εντολή – χαρακτήρα "D" προς το πρόγραμμα. Αν δοθεί η εντολή – χαρακτήρας "D" διαβάζεται ένας μόνο χαρακτήρας από τα δεδομένα. Τα αποτελέσματα του προγράμματος σε κάθε περίπτωση παρουσιάζονται με κατάλληλο τρόπο σε οθόνη τερματικού. Το πρόγραμμα επιστρέφει πάντα στην ερώτηση:

GIVE COMMAND CHARACTER

και σταματά μόνο αν πληκτρολογηθεί CTL-C.

Κείμενο προγράμματος: Στη συνέχεια παρατίθεται το πρόγραμμα **CONTRL** σε γλώσσα προγραμματισμού FORTRAN/9000, όπως αυτή υπήρχε στα συστήματα H/Y HP-9000/320-370 (βλ. και Hewlett Packard, 1987), που αναπτύχθηκε και χρησιμοποιήθηκε από τον Σιμόπουλο (1994) για περιβάλλον Λ/Σ HP-UX 6.x και 7.x. Το πρόγραμμα αποτελείται από ένα αρχείο **.f**. Μετά το πρόγραμμα **CONTRL** παρατίθενται τα αρχεία **opendv.c**, **modedv.c**, τα οποία περιέχουν τις κατάλληλες εκδόσεις των υπορουτινών **opencnt** και **setcnt** της βιβλιοθήκης **count**, ώστε να υποστηρίζεται το canonical mode.

[3] Μοναδικό χαρακτήρα σύμφωνα με τα πρωτόκολλα επικοινωνίας των μονάδων επικοινωνίας 1788 ή 2088

contrl.f

```
$INCLUDE 'count.f.i'

      PROGRAM CONTRL
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.HARVATIS  April 88
c     It also works with 2088 ; SES/Feb-94
      BYTE KI,K40,K41,M,N,KD
      DATA KI,K40,K41/1HI,1H  ,1H!/,KD/1HD/
      CALL OPENCNT(IDV,0,"/dev/tty05"//char(0))
   3  PRINT 1
   1  FORMAT('$ GIVE COMMAND CHARACTER :')
      READ 2,N
   2  FORMAT(A1)
      IF(N.EQ.KI)GO TO 5
      IF(N.EQ.KD)GO TO 11
      CALL TOCNT(IDV,N)
      GO TO 3
  11  CALL FMCNT(IDV,M)
      PRINT 4,M,M
      GO TO 3
   5  CALL FMCNT(IDV,M)
      PRINT 4,M,M
   4  FORMAT(1X,'M= ',A1,' Octal :',O16)
      IF(M.EQ.K40)GO TO 8
      IF(M.EQ.K41)GO TO 8
      PRINT 7,M,K40,K41
      PRINT 17,M,K40,K41
  17  FORMAT(1X,O16,'---',O16,'---',O16)
   7  FORMAT(' ERROR , M,K40,K41  ',3(A2,2X))
   8  CALL TOCNT(IDV,N)
      CALL DATAIN(IDV,IC,ICNT)
      PRINT 15,IC
  15  FORMAT(I4)
      PRINT 6,ICNT
   6  FORMAT(1X,I6)
      GO TO 3
      END
```

opendv.c

```
/* AUTHOR : D. Harvatis  - March 88 */
/* Modified for CI 2088 by SES Feb-94 */

#include <errno.h>
#include <unistd.h>
#include <fcntl.h>
#include <termio.h>
#include <stdio.h>

extern void SetCnt(); /* SetCnt(dv,cnt,t) int dv,cnt,t; */

void ClearCnt(dv)
int dv;
{
      if ( ioctl(dv,TCFLSH,2) == -1) {
            fprintf(stderr,"ClearCnt : ioctl failed.\n");
            exit(1);
      }
}

void OpenCnt(dv,cnt,df)
int *dv,*cnt;
char *df;
{
      int loc;
      if ((*dv = open(df,O_RDWR)) < 0) {
            if (errno==ENAMETOOLONG || errno==ENOTDIR
                  || errno==ENOENT || errno==EISDIR )
                  fprintf(stderr,
                  "\nOpenCnt : bad file name.\n");
            else if (errno==EMFILE || errno==ENFILE)
                  fprintf(stderr,
                  "\nOpenCnt : too many open files.\n");
            else if (errno==EACCES || errno==EROFS || errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : permission denied.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : open failed.\n");
            exit(1);
      }
      /* check if RS-232 is used by another program */
      loc=lockf(*dv,F_TLOCK,0);
      if (loc != 0) {
            if (errno==EACCES || errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : line unavailable - try again
later.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : unexpected lockf error.\n");
            exit(1);
      }
      SetCnt(*dv,*cnt,-1);
      ClearCnt(*dv);
}

void CloseCnt(dv)
int dv;
```

```
{
      if (close(dv) != 0) {
            fprintf(stderr,"CloseCnt : close failed.\n");
            exit(1);
      }
}
```

modedv.c

```
/* AUTHOR : D. Harvatis  - March 88  */
/* Modified for CI 2088 by SES Feb-94 */

#include <termio.h>
#include <stdio.h>

void SetCnt(dv,cnt,t)
int dv,cnt,t;  /* 0:1788, 1:2088  */
{
      unsigned char c;
      struct termio arg;
      if (t != 0)
            c = 1;
      else
            c = 0;
      if (cnt == 0){
        arg.c_iflag = IGNPAR;
        arg.c_lflag = 0;}
      else {
        arg.c_iflag = IGNPAR|ICRNL;
        arg.c_lflag = ICANON;}
      arg.c_oflag = 0;
      arg.c_cflag = B9600|CS7|CREAD|PARENB|HUPCL;
      arg.c_line = 0;
      arg.c_cc[VINTR] = CINTR;
      arg.c_cc[VQUIT] = CQUIT;
      arg.c_cc[VERASE] = CERASE;
      arg.c_cc[VKILL] = CKILL;
      arg.c_cc[VMIN] = c;
      arg.c_cc[VTIME] = 0;
      arg.c_cc[VEOF] = CSWTCH;
      if (ioctl(dv,TCSETA,&arg) == -1) {
            fprintf(stderr,"\nSetCnt : ioctl failed.\n");
            exit(1);
      }
}
```

5.4.6 Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας

Στη συνέχεια παρατίθενται μετασκευή του πιοπάνω προγράμματος **CONTRL** σε γλώσσα προγραμματισμού FORTRAN77, το οποίο βασίζεται στο αντίστοιχο του Σιμόπουλου (1994). Το πρόγραμμα αποτελείται από ένα αρχείο **.F**, όπως στην προηγούμενη ενότητα και προσαρμόσθηκε αποκλειστικά για να συνεργάζεται μόνο με τη μονάδα επικοινωνίας 2088 ώστε να διαβάζει τυπικές ακολουθίες χαρακτήρων, όπως παράγονται από την επικοινωνία Η/Υ – μονάδας 2088 – καταμετρητών 1776 σε σύνδεση Daisy Chain. Πραγματοποιήθηκαν φυσικά και αλλαγές για την μετατροπή του από FORTRAN/9000 σε FORTRAN77. Το πρόγραμμα χρησιμοποιεί την βιβλιοθήκη **count,** που παρουσιάσθηκε στο προηγούμενο 4^{ο} Κεφάλαιο. Η ανάγνωση δεδομένων μέσω της μονάδας επικοινωνίας 2088 γίνεται χαρακτήρα – χαρακτήρα σε non-canonical mode. Το πρόγραμμα λειτουργεί στο Λ/Σ LINUX / POSIX, έκδοση Fedora Core 8. Η καλή λειτουργία του προγράμματος δοκιμάσθηκε με επιτυχία σε αυτήν την ΔΕ στην πειραματική διάταξη της εδώ ενότητας (§ 5.4).

Το πρόγραμμα με το εδώ όνομα **P20887** μπορεί να χρησιμοποιείται –όπως και το προηγούμενο πρόγραμμα **CONTRL**, για τον έλεγχο των διατάξεων counter-timer και για τη συλλογή των μετρήσεών τους, τις οποίες αποθηκεύει σε αρχεία. Η περιγραφή και οι οδηγίες χρήσεως του προγράμματος έχουν χονδρικά ακριβώς όπως στην προηγούμενη παράγραφο.

Σχόλια: Το πρόγραμμα **P20887** τρέχει με την απλή εντολή:

./P20887

(αρκεί το αρχείο στο οποίο περιέχεται το προκύπτον εκτελέσιμο να έχει αυτό το όνομα και επίσης να έχει γίνει με ορθό τρόπο η μετάφραση και η σύνδεση).

Κείμενο προγράμματος: Στη συνέχεια παρατίθενται το πρόγραμμα **P20887** σε γλώσσα προγραμματισμού FORTRAN77, όπως αυτή υπάρχει σε σύστημα Η/Υ τύπου PC συμβατού για περιβάλλον Λ/Σ LINUX / POSIX έκδοση Fedora Core 8. Το πρόγραμμα αποτελείται από ένα αρχείο **.F.**.

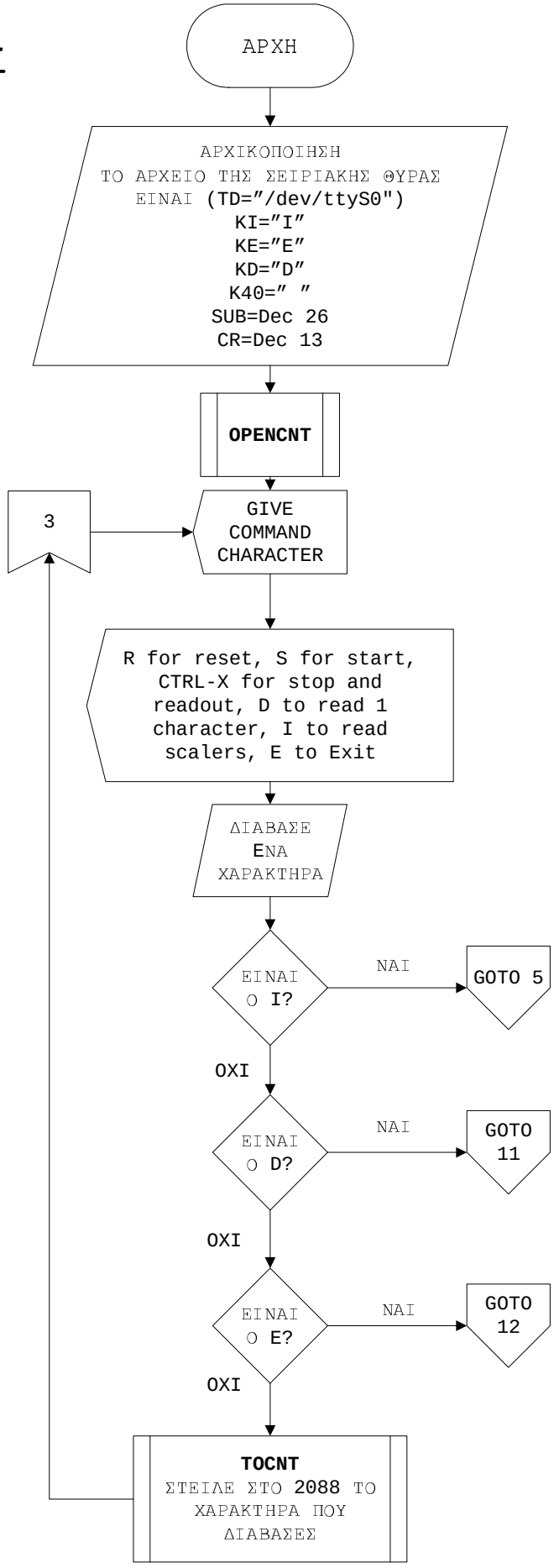
P20887.F

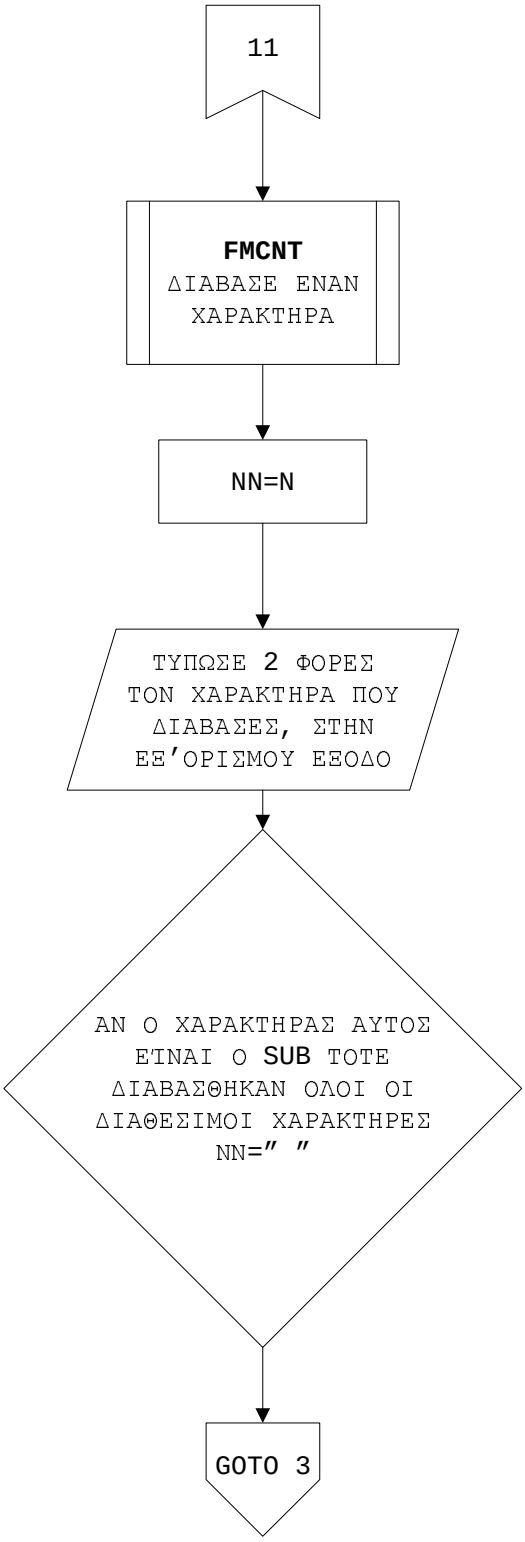
```
#include "count.f.i"
      PROGRAM P20887
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.HARVATIS  April 88
C     It also works with 2088 ; SES/Feb-94
C     Fedora Core 8 version for 2088 by G.D Alexopoulos May-08
      CHARACTER*1 M,N,NN,CR,KI,K40,KD,KE,SUB
      DATA KI,KE,K40,KD/"I","E"," ","D"/
      CHARACTER*20 TD
      TD="/dev/ttyS0"
      CR=CHAR(13)
      SUB=CHAR(26)
      NN=" "
      CALL OPENCNT(IDV,TD)
   3  PRINT 1
   1  FORMAT('$ GIVE COMMAND CHARACTER :')
      WRITE(*,*) " R for reset, S for start "
      WRITE(*,*) " CTRL-X for stop and readout "
      WRITE(*,*) " D to read 1 char, I to read scalers, E to exit"
      READ(*,2) N
      IF(N.EQ.KI)GO TO 5
      IF(N.EQ.KD)GO TO 11
      IF(N.EQ.KE)GO TO 12
      CALL TOCNT(IDV,N)
      GO TO 3
  11  CALL FMCNT(IDV,M)
      NN=N
      PRINT 4,M,M
      IF (M.EQ.SUB) NN=" "
      GO TO 3
   5  IF(NN.NE."D") THEN
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
   7  CALL FMCNT(IDV,M)
      PRINT 4,M,M
      ELSE
      WRITE (*,*) " YOU ARE NOT AT THE 1ST CHARACTER OF DATA CHAIN "
      GO TO 12
      ENDIF
      CALL DATAIN(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      CALL DATAIN(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      IF (M.EQ.SUB) GO TO 3
      IF (M.EQ.K40) GO TO 7
  12  CALL CLOSECNT(IDV)
      WRITE(*,*) " PROGRAM WILL END IN 2 SEC"
      CALL SLEEP(2)
   2  FORMAT(A1)
   4  FORMAT(1X,'M= ',A1,' Octal :',O16)
   6  FORMAT(1X,I6)
```

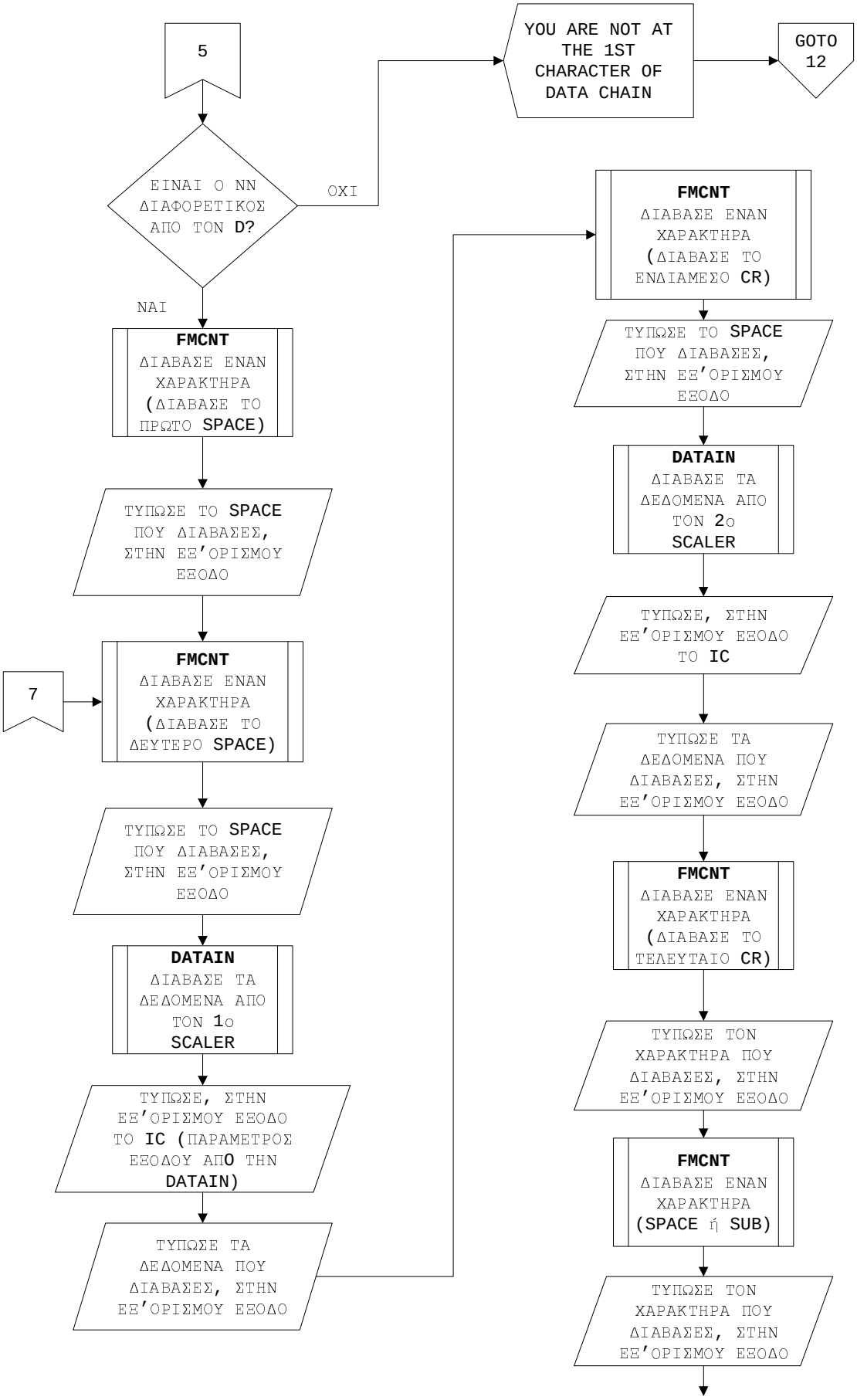
```
15  FORMAT(I4)
    STOP
    END
```

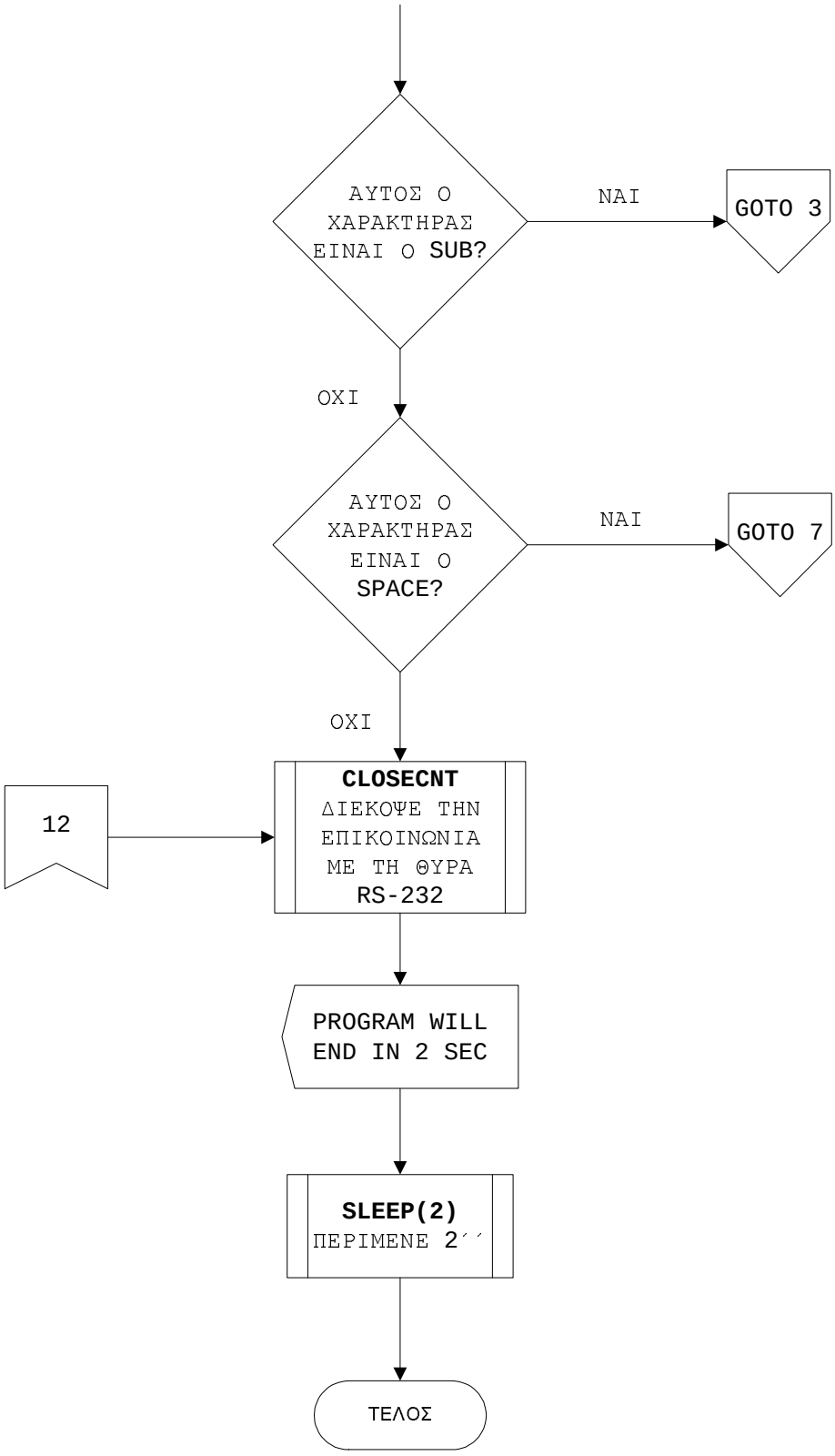
Σημειώνεται ότι το πρόγραμμα **P20887** που παρουσιάζεται στην εδώ § 5.4.6 περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία. Το λογικό διάγραμμα του προγράμματος **P20887** για την προγραμματιστική υλοποίηση της παρούσης ΔΕ, παρουσιάζεται στη συνέχεια.

P20887.F for 2088




11
FMCNT
ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ
NN=N
ΤΥΠΩΣΕ 2 ΦΟΡΕΣ ΤΟΝ ΧΑΡΑΚΤΗΡΑ ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
ΑΝ Ο ΧΑΡΑΚΤΗΡΑΣ ΑΥΤΟΣ ΕΙΝΑΙ Ο SUB ΤΟΤΕ ΔΙΑΒΑΣΘΗΚΑΝ ΟΛΟΙ ΟΙ ΔΙΑΘΕΣΙΜΟΙ ΧΑΡΑΚΤΗΡΕΣ NN=" "
GOTO 3


5
ΕΙΝΑΙ Ο NN ΔΙΑΦΟΡΕΤΙΚΟΣ ΑΠΟ ΤΟΝ D?
ΟΧΙ
YOU ARE NOT AT THE 1ST CHARACTER OF DATA CHAIN
GOTO 12
ΝΑΙ
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΔΙΑΒΑΣΕ ΤΟ ΠΡΩΤΟ SPACE)
ΤΥΠΩΣΕ ΤΟ SPACE ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
7
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΔΙΑΒΑΣΕ ΤΟ ΔΕΥΤΕΡΟ SPACE)
ΤΥΠΩΣΕ ΤΟ SPACE ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
DATAIN ΔΙΑΒΑΣΕ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΟΝ 1ο SCALER
ΤΥΠΩΣΕ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ ΤΟ IC (ΠΑΡΑΜΕΤΡΟΣ ΕΞΟΔΟΥ ΑΠΟ ΤΗΝ DATAIN)
ΤΥΠΩΣΕ ΤΑ ΔΕΔΟΜΕΝΑ ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΔΙΑΒΑΣΕ ΤΟ ΕΝΔΙΑΜΕΣΟ CR)
ΤΥΠΩΣΕ ΤΟ SPACE ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
DATAIN ΔΙΑΒΑΣΕ ΤΑ ΔΕΔΟΜΕΝΑ ΑΠΟ ΤΟΝ 2ο SCALER
ΤΥΠΩΣΕ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ ΤΟ IC
ΤΥΠΩΣΕ ΤΑ ΔΕΔΟΜΕΝΑ ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (ΔΙΑΒΑΣΕ ΤΟ ΤΕΛΕΥΤΑΙΟ CR)
ΤΥΠΩΣΕ ΤΟΝ ΧΑΡΑΚΤΗΡΑ ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ
FMCNT ΔΙΑΒΑΣΕ ΕΝΑΝ ΧΑΡΑΚΤΗΡΑ (SPACE ή SUB)
ΤΥΠΩΣΕ ΤΟΝ ΧΑΡΑΚΤΗΡΑ ΠΟΥ ΔΙΑΒΑΣΕΣ, ΣΤΗΝ ΕΞ'ΟΡΙΣΜΟΥ ΕΞΟΔΟ


ΑΥΤΟΣ Ο ΧΑΡΑΚΤΗΡΑΣ ΕΙΝΑΙ Ο SUB?
ΝΑΙ
GOTO 3
ΟΧΙ
ΑΥΤΟΣ Ο ΧΑΡΑΚΤΗΡΑΣ ΕΙΝΑΙ Ο SPACE?
ΝΑΙ
GOTO 7
ΟΧΙ
12
CLOSECNT
ΔΙΕΚΟΨΕ ΤΗΝ ΕΠΙΚΟΙΝΩΝΙΑ ΜΕ ΤΗ ΘΥΡΑ RS-232
PROGRAM WILL END IN 2 SEC
SLEEP(2)
ΠΕΡΙΜΕΝΕ 2΄΄
ΤΕΛΟΣ

5.4.7 Μετάφραση και σύνδεση

Λαμβάνοντας υπόψιν τα όσα σημειώθηκαν στη παράγραφο 4.8 και δεδομένου ότι το πρόγραμμα **2088.7** χρησιμοποιεί για να λειτουργήσει τις υπορουτίνες της βιβλιοθήκης **count**, η μετάφραση και η σύνδεσή του για την παραγωγή του αντίστοιχου εκτελέσιμου **radon** θα μπορούσε να γίνει από την εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –g *P20887*.F –l count –o *P20887*

η οποία μπορεί να αποτελεί και μέρος άλλου ευρύτερου makefile.

5.5 Καταμετρητής 2071A/w03 με μονάδα επικοινωνίας 207x-03

5.5.1 Γενικά

Ο καταμετρητής 2071A/w03 με ενσωματωμένη μονάδα επικοινωνίας 207x-03 που θα χρησιμοποιηθεί έχει σειριακό αριθμό 388899 και διακρίνεται στις φωτογραφίες των σχημάτων 2.1 (εμπρόσθια όψη) και 2.2 (οπίσθια όψη) του αντίστοιχου εγχειριδίου (βλ. και Παράρτημα). Στη φωτογραφία του Διαγράμματος 5.12, διακρίνεται η πραγματική πίσω όψη του καταμετρητή 2071A/w03, με την ενσωματωμένη σειριακή θύρα της μονάδας 207x-03. Στη συνέχεια παρουσιάζονται τα εξωτερικά λειτουργικά μέρη του συστήματος αυτού που είναι απαραίτητα για την παρούσα ΔΕ. Τα λοιπά λειτουργικά μέρη αγνοούνται. Έτσι λοιπόν:

Στο εμπρόσθιο μέρος υπάρχουν

- διάφοροι διακόπτες για τη ρύθμιση της λειτουργίας του καταμετρητή 2071A/w03, οι οποίοι δεν έχουν άμεση σχέση με την επικοινωνία που εξετάζεται
- διακόπτης SINGLE/RECYCLE. Για να λειτουργεί το σύστημα κατά τον τρόπο που είναι επιθυμητός, ο διακόπτης αυτός πρέπει να είναι στη θέση SINGLE

Στο πίσω μέρος υπάρχουν

- μία θύρα (EIA DATA) επικοινωνιών τύπου RS232C female 25 pin (η μονάδα χρησιμοποιείται ως εξοπλισμός DTE).

5.5.2 Ρύθμιση

Υπενθυμίζεται και πάλι ότι, κατά την κατάστρωση του λογισμικού, οι υπορουτίνες ρυθμίσεως της σειριακής θύρας RS-232 του Η/Υ, γράφθηκαν ώστε να τηρούνται οι ακόλουθες προδιαγραφές:

Ταχύτητα Μετάδοσης 9600 baud.

Μήκος χαρακτήρα 7 bits.

Parity EVEN.

Ένα stop bit

Όπως πληροφορεί το εγχειρίδιο της συσκευής (βλ. και Operator's Manual 207x-03, 1988), αυτή όπως έρχεται προρυθμισμένη από το εργοστάσιο κατασκευής έχει

(α) **μήκος χαρακτήρα 8 bits, NO Parity, Δύο stop bit (ρυθμίσεις που δεν μπορούν να αλλάξουν)**

και

(β) τους μικροδιακόπτες (στα αγγλικά dip switches)που βρίσκονται στο πλευρό της (βλ. και φωτογραφία στο Διάγραμμα 5.13) στις θέσεις:

1) Μικροδιακόπτης 1 (SW1) στη θέση OFF
2) Μικροδιακόπτης 2 (SW2) στη θέση OFF
3) Μικροδιακόπτης 3 (SW3) στη θέση ON
4) Μικροδιακόπτης 4 (SW4) στη θέση ON
5) Μικροδιακόπτης 5 (SW5) στη θέση ON
6) Μικροδιακόπτες SW6 – SW8 στις θέσεις 6:OFF, 7:OFF, 8:ON
7) Μικροδιακόπτης 9 (SW9) στη θέση OFF
8) Μικροδιακόπτης 10 (SW10) στη θέση OFF

Προφανώς, πριν την σύνδεση της συσκευής έγινε ο απαραίτητος έλεγχος των μικροδιακοπτών ώστε αυτή να είναι συμβατή με τις ρυθμίσεις της σειριακής θύρας RS-232 του Η/Υ. Για το σκοπό αυτό ρυθμίσθηκαν τα ακόλουθα:

1) Μικροδιακόπτης 1 (SW1) στη θέση OFF (σημαίνει ότι ο καταμετρητής λειτουργεί "Single Cycle Mode", δηλ. μετά το τέλος μίας μέτρησης, αποστέλλει τα δεδομένα που έχουν καταγραφεί στους scalers στην σειριακή θύρα και δεν ξαναξεκινά εκτός αν ο διακόπτης SINGLE/RECYCLE είναι στη θέση RECYCLE)
2) Μικροδιακόπτης 2 (SW2) στη θέση OFF (σημαίνει ότι δεν υπάρχουν form feed χαρακτήρες στην ακολουθία των δεδομένων που αποστέλλονται στη σειριακή θύρα)

3) Μικροδιακόπτης 3 (SW3) στη θέση OFF (σημαίνει ότι δεν υπάρχουν line feed χαρακτήρες στην ακολουθία των δεδομένων που αποστέλλονται στη σειριακή θύρα)

4) Μικροδιακόπτης 4 (SW4) στη θέση OFF (σημαίνει ότι δεν υπάρχουν carriage return χαρακτήρες στην ακολουθία των δεδομένων που αποστέλλονται στη σειριακή θύρα)

5) Μικροδιακόπτης 5 (SW5) στη θέση OFF (σημαίνει ότι δεν υπάρχουν space χαρακτήρες στην ακολουθία των δεδομένων που αποστέλλονται στη σειριακή θύρα)

6) Μικροδιακόπτες SW6 – SW8 στις θέσεις 6:ON, 7:ON, 8:OFF (σημαίνει ότι η ταχύτητα επικοινωνίας είναι 9600 BAUD)

7) Μικροδιακόπτης 9 (SW9) στη θέση OFF (η ρύθμισή του έχει σημασία μόνο αν ο μικροδιακόπτης 10 (SW10) είναι στη θέση ON)

8) Μικροδιακόπτης 10 (SW10) στη θέση OFF (σημαίνει ότι ο καταμετρητής 2071A/w03 είναι μοναδικός – Single Unit Mode, δηλαδή δεν υπάρχουν άλλοι καταμετρητές συνδεδεμένοι με την ίδια σειριακή θύρα της ενσωματωμένης μονάδας επικοινωνίας 207x-03).

Επισημαίνεται στο σημείο αυτό, ότι ο καταμετρητής 2071A/w03 με την ενσωματωμένη μονάδα επικοινωνίας 207x-03, θα εξετασθεί στα πλαίσια της παρούσας Διπλωματικής Εργασίας, μόνο σε Single Unit Mode, κυρίως διότι το ΕΠΤ-ΕΜΠ δεν διαθέτει δεύτερη παρόμοια μονάδα και βέβαια διότι, οι μονάδες αυτού του τύπου δεν κατασκευάζονται πλέον.

Ο καταμετρητής 2071A/w03 με την ενσωματωμένη μονάδα επικοινωνίας 207x-03, με τις πιοπάνω ρυθμίσεις μπορεί να χρησιμοποιείται στο ΕΠΤ-ΕΜΠ. Σημειώνεται ότι η μονάδα επικοινωνίας 207x-03 είναι κατασκευασμένη έτσι ώστε να "διαβάζει" ψηφιακούς καταχωρητές από καταμετρητές 207x μήκους οκτώ χαρακτήρων.

5.5.3 Πρωτόκολλο επικοινωνίας

Η μονάδα επικοινωνίας διαθέτει πρωτόκολλο σειριακής σύνδεσης μέσω του οποίου η σειριακή θύρα του συνδεδεμένου Η/Υ είναι δυνατόν να ελέγχει έναν καταμετρητή και να παραλαμβάνει δεδομένα από αυτόν. Το πρωτόκολλο αυτό έχει δύο συνιστώσες, το πρωτόκολλο αποστολής και το πρωτόκολλο λήψης, ως εξής: (α)

Αποστολή χαρακτήρων ASCII ≡ Εντολές ελέγχου από τον Η/Υ προς τη μονάδα επικοινωνίας 207x-03, και (β) Λήψη χαρακτήρων ASCII ≡ Πληροφορίες από την μονάδα επικοινωνίας προς τον Η/Υ. Τα δύο πρωτόκολλα είναι (όπως δίνονται από το αντίστοιχο εγχειρίδιο, βλ. και Παράρτημα):

Πρωτόκολλο αποστολής από τον Η/Υ

ASCII Code (Hex)	**Χαρακτήρας**	**Ορισμός**
18	CAN	(CTL-X key) clear unit, start counting
19	EM	(CTL-Y) Stop unit
17	ETB	(CTL-W) Readout unit (counting or stopped)

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που δέχεται η μονάδα 207x-03 από τον Η/Υ ερμηνεύονται ως εντολές που αναλύονται ως εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
24	CAN	όταν ο Η/Υ αποστέλλει ένα CAN η μονάδα επικοινωνίας 207x-03 αποστέλλει ένα σήμα RESET / START στον καταμετρητή, με αποτέλεσμα όλοι οι scalers να μηδενίσουν και αυτόματα μετά το μηδενισμό να αρχίσει νέα μέτρηση.
25	EM	όταν ο Η/Υ αποστέλλει ένα EM η μονάδα επικοινωνίας 207x-03 σταματάει τη μέτρηση.
23	ETB	όταν ο Η/Υ αποστέλλει ένα ETB η μονάδα επικοινωνίας 207x-03 αποστέλλει το περιεχόμενο των scalers στη σειριακή θύρα του Η/Υ.

Προσοχή χρειάζεται στην περίπτωση που ο καταμετρητής έχει ολοκληρώσει μία μέτρηση διότι ολοκληρώθηκε ο προρυθμισμένος για αυτήν χρόνος. Τότε το

περιεχόμενο των scalers αποστέλλεται αυτόματα στη σειριακή θύρα του Η/Υ, χωρίς να μεσολαβήσει εντολή CTL-W.

Πρωτόκολλο λήψης προς τον Η/Υ

ASCII Code (octal)	**Χαρακτήρας**	**Ορισμός**
060 έως 067	0 έως 7	Data from module
070, 071	8, 9	Data from module
19	DC3	όταν η μονάδα 207x-03 αποστείλει ένα "XOFF" ο Η/Υ σταματάει να αποστέλλει δεδομένα (εντολές) προς την μονάδα 207x-03. Δεδομένου ότι προγραμματιστικά το πρωτόκολλο XON – XOFF στην εδώ περίπτωση είναι απενεργοποιημένο, η αποστολή XOFF από τη μονάδα 207x-03 είναι χωρίς σημασία.
17	DC1	όταν η μονάδα 207x-03 αποστείλει ένα "XON" ο Η/Υ μπορεί να αποστέλλει δεδομένα (εντολές) προς την μονάδα 207x-03. Δεδομένου ότι προγραμματιστικά το πρωτόκολλο XON – XOFF στην εδώ περίπτωση είναι απενεργοποιημένο, η αποστολή XOFF από τη μονάδα 207x-03 είναι χωρίς σημασία.

Σύμφωνα με το πρωτόκολλο αυτό, οι χαρακτήρες που αποστέλλει η μονάδα 207x-03 στον Η/Υ είναι μόνο πληροφορίες που σημαίνουν τα εξής:

ASCII Code (decimal)	**Χαρακτήρας**	**Ορισμός**
19	DC3	(XOFF) Αρχικός χαρακτήρας μίας

		δεκαεξαψήφιας ένδειξης (= δύο οκταψήφιες ενδείξεις από τους scalers) ή
19	DC3	(XOFF) Χαρακτήρας αναγνώρισης ότι λήφθηκε μία εντολή από τον H/Y και πρέπει να επεξεργασθεί από τη μονάδα 207x-03 (π.χ. CTL-X, CTL-Y ή CTL-W).
17	DC1	(XON) Τερματικός χαρακτήρας μίας δεκαεξαψήφιας ένδειξης (= δύο οκταψήφιες ενδείξεις από τους scalers) ή
17	DC1	(XON) Χαρακτήρας αναγνώρισης ότι εκτελέσθηκε μία εντολή που λήφθηκε από τον H/Y.
48 έως 57	0 έως 9	Ψηφία που κατά περίπτωση συγκροτούν το περιεχόμενο δύο οκταψήφιων ενδείξεων από τους scalers

Είναι πολύ σημαντικό να παρατηρηθεί εδώ ότι η μονάδα επικοινωνίας 207x-03, μπορεί να έχει ως μονάδα μέτρησης δεδομένων τόσο τη λεγόμενη "λέξη" (των οκτώ χαρακτήρων), διότι αν οι σχετικοί μικροδιακόπτες ρυθμισθούν κατάλληλα, οι λέξεις μπορούν να διαχωρίζονται με CR ή NL. Αυτό πρακτικά σημαίνει ότι η ανάγνωση των λέξεων που μεταδίδονται προς τον H/Y από τη μονάδα 207x-03 δια μέσου της σειριακής θύρας RS-232, μπορεί να γίνεται είτε χαρακτήρα – χαρακτήρα σε non-canonical mode, είτε λέξη – λέξη σε canonical mode.

Σημειώνεται επίσης ότι η μονάδα 207x-03 μπορεί να χρησιμοποιήσει το πρωτόκολλο XON – XOFF για επικοινωνία με την σειριακή θύρα ενός H/Y. Οι χαρακτήρες αυτοί στην εδώ περίπτωση μπορούν να σταλούν από την μονάδα207x-03 προς την σειριακή θύρα του H/Y. Επισημαίνεται βέβαια ότι στις υπορουτίνες ρύθμισης της σειριακής θύρας RS-232 του H/Y, που συμπεριλαμβάνονται στην βιβλιοθήκη count, το πρωτόκολλο XON – XOFF είναι απενεργοποιημένο.

Δίνεται μια τυπική ακολουθία χαρακτήρων, όπως παράγεται από την επικοινωνία Η/Υ – μονάδας 207x-03 – ενός καταμετρητή 2071A μετά την αποστολή των δεδομένων από τους scalers στον Η/Υ ή μετά την εκτέλεση μίας εντολής CTL-W (τα κόμματα δεν μετράνε στην ακολουθία):

XOFF,11111111,22222222,XON

Επίσης δίνεται μια τυπική ακολουθία χαρακτήρων, όπως παράγεται από την επικοινωνία Η/Υ – μονάδας 207x-03 – ενός καταμετρητή 2071A μετά την αποστολή μίας εντολής CTL-X ή CTL-Y από τον Η/Υ προς τη μονάδα 207x-03.

XOFF,XON

Τέλος, όταν εκκινεί η μονάδα επικοινωνίας 207x-03, αποστέλλεται προς τη σειριακή θύρα του Η/Υ η ακολουθία χαρακτήρων:

,,SP,E,I,A,SP,2,.,6,SP,*,*

όπου

SP Space, δηλ. κενό

Πέρα από το πρωτόκολλο που αναλύθηκε υπάρχουν και άλλες εντολές προς την μονάδα επικοινωνίας 207x-03, τις οποίες ο αναγνώστης μπορεί να μελετήσει στη σελ. 22 του σχετικού εγχειριδίου χρήσης (βλ. και Παράρτημα).

Σύμφωνα με αυτό το λογικό διάγραμμα στην έναρξη της διαδικασίας το αντίστοιχο πρόγραμμα περιμένει να τελειώσει η τρέχουσα μέτρηση και να αποσταλούν τα δεδομένα προς τη σειριακή θύρα του Η/Υ. Η μέτρηση μπορεί να τελειώσει αν εξαντληθεί ο προρυθμισμένος χρόνος μέτρησης στους καταμετρητές ή αν πατηθεί το MANUAL PRINT button,ή αν πατηθεί οποιοδήποτε STOP button σε οποιοδήποτε από τους συνδεδεμένους καταμετρητές, ή αν ληφθεί CAN από τον Η/Υ προς την μονάδα 2088. Για την υλοποίηση του υπόψιν λογικού διαγράμματος υποτίθεται ότι ο Η/Υ είναι ικανός να δέχεται δεδομένα με αρκετή ταχύτητα, χωρίς να χάνει χαρακτήρες.

5.5.4 Σύνδεση με την πειραματική διάταξη

Μετά τη ρύθμιση ο υπόψιν καταμετρητής με τη μονάδα επικοινωνίας τοποθετήθηκε στην μονάδα NIM Bin. Ο καταμετρητής συνδέθηκε με την σειριακή θύρα του Η/Υ στη θέση "FROM" του σχετικού κυκλώματος υποκλοπής (βλ. και φωτογραφία του Διαγράμματος 5.14) με σειριακό καλώδιο το οποίο κατασκευάσθηκε στο πλαίσιο της

παρούσης ΔΕ, σύμφωνα με τις οδηγίες του σχετικού εγχειριδίου Operator's Manual 207x-03 (1988), σελ. 24. Προτείνεται το ειδικό αυτό καλώδιο να παραμένει πάντοτε τοποθετημένο πάνω στον καταμετρητή 2071A, ώστε να μην χαθεί. Σχηματική αναπαράσταση του καλωδίου διακρίνεται στο σχήμα του Διαγράμματος 5.15.

5.5.5 Προγραμματιστική υλοποίηση από τον Σιμόπουλο (1997)

Ένα ολοκληρωμένο παράδειγμα χρησιμοποίησης του καταμετρητή 2071A/w03 και της μονάδας επικοινωνίας 207x-03, αναπτύχθηκε από τον Σιμόπουλο (1997) στη βάση των σχετικών προγραμμάτων σε γλώσσες Microsoft FORTRAN και assembly για επεξεργαστές Intel 80x86, που διατύπωσε ο Βοσυνιώτης (1990). Τα σχετικά κείμενα προγραμμάτων βρίσκονται στο Παράρτημα. Κώδικας χρησιμοποίησης του καταμετρητή 2071A/w03 και της μονάδας επικοινωνίας 207x-03 σε περιβάλλον Λ/Σ UNIX ή LINUX δεν είχε αναπτυχθεί πριν την παρούσα Διπλωματική Εργασία.

5.5.6 Προγραμματιστική υλοποίηση Διπλωματικής Εργασίας

Στη συνέχεια παρατίθενται μετασκευή του προγράμματος **P20887** σε γλώσσα προγραμματισμού FORTRAN77. Το πρόγραμμα αποτελείται από ένα αρχείο **.F**, όπως στην προηγούμενη παράγραφο 5.4.6 και προσαρμόσθηκε αποκλειστικά για να συνεργάζεται μόνο με τον καταμετρητή 2071A και τη μονάδα επικοινωνίας 207x-03 ώστε να διαβάζει τυπικές ακολουθίες χαρακτήρων, όπως παράγονται από την επικοινωνία Η/Υ – μονάδας 207x-03 – καταμετρητή 2071A. Το πρόγραμμα χρησιμοποιεί την βιβλιοθήκη **count,** που παρουσιάσθηκε στο προηγούμενο 4^{ο} Κεφάλαιο, επαυξημένη με υπορουτίνες που περιέχονται σε αρχεία με το όνομα **open207dv.c**, **mode207dv.c**, **data8dv.c** και **onion207.F**. Οι νέες υπορουτίνες χρειάζονται για δύο λόγους: (α) διότι η σειριακή επικοινωνία στην περίπτωση της μονάδας 207x-03 γίνεται 8 bit, No Parity, 2 stop bits (αντί για 7 bit, EVEN Parity, 1 stop bit) και (β) διότι οι scalers του καταμετρητή 2071A έχουν 8 χαρακτήρες (αντί για 6 που έχουν οι scalers του καταμετρητή 1776). Οι αλλαγές που έχουν οι νέες υπορουτίνες σε σχέση με τις αντίστοιχες του 4^{ου} Κεφαλαίου σημειώνονται με κατάλληλη σκίαση [ ]. Η ανάγνωση δεδομένων μέσω της μονάδας επικοινωνίας 207x-03 γίνεται χαρακτήρα – χαρακτήρα σε non-canonical mode. Το πρόγραμμα λειτουργεί στο Λ/Σ LINUX / POSIX, έκδοση Fedora Core 8. Η καλή λειτουργία του προγράμματος δοκιμάσθηκε με επιτυχία σε αυτήν την ΔΕ στην πειραματική διάταξη της εδώ ενότητας (§ 5.5).

Το πρόγραμμα με το εδώ όνομα **P20718** μπορεί να χρησιμοποιείται –όπως και το προηγούμενο πρόγραμμα **P20887**, για τον έλεγχο των διατάξεων counter-timer και για τη συλλογή των μετρήσεών τους, τις οποίες αποθηκεύει σε αρχεία. Η περιγραφή και οι οδηγίες χρήσεως του προγράμματος είναι πολύ απλές και βασίζονται στο πρωτόκολλο επικοινωνίας που παρουσιάσθηκε.

Σχόλια: Το πρόγραμμα **P20718** τρέχει με την απλή εντολή:

./P20718

(αρκεί το αρχείο στο οποίο περιέχεται το προκύπτον εκτελέσιμο να έχει αυτό το όνομα και επίσης να έχει γίνει με ορθό τρόπο η μετάφραση και η σύνδεση).

Κείμενο προγράμματος: Στη συνέχεια παρατίθεται το πρόγραμμα **P20718** σε γλώσσα προγραμματισμού FORTRAN77, όπως αυτή υπάρχει σε σύστημα Η/Υ τύπου PC συμβατού για περιβάλλον Λ/Σ LINUX / POSIX έκδοση Fedora Core 8. Το πρόγραμμα αποτελείται από ένα αρχείο **.F**. Επίσης παρατίθεται το πρόγραμμα **onion207.F**, και τα προγράμματα σε γλώσσα προγραμματισμού C **open207dv.c**, **mode207dv.c** και **data8dv.c**

P20718.F

```
#include "count.f.i"
      PROGRAM P20718
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.HARVATIS  April 88
C     Fedora Core 8 version for 207x-03 by G.D Alexopoulos May-08
      CHARACTER*1 M,N,NN,CW,KI,KD,KE,DC1
      DATA KI,KE,K40,KD/"I","E"," ","D"/
      CHARACTER*20 TD
      TD="/dev/ttyS0"
      DC1=CHAR(17)
      CW=CHAR(23)
      NN=" "
      CALL OPENCNT2071(IDV,TD)
   3  PRINT 1
   1  FORMAT('$ GIVE COMMAND CHARACTER :')
      WRITE(*,*) " Ctrl-X for reset and start, Ctrl-Y to stop "
      WRITE(*,*) " Ctrl-W to readout "
      WRITE(*,*) " D to read 1 char, I to read scalers "
      WRITE (*,*) " to type Ctrl-W, type CTRL-V CTRL-W "
      READ(*,2) N
      IF(N.EQ.KI)GO TO 5
      IF(N.EQ.KD)GO TO 11
      IF(N.EQ.KE)GO TO 12
      CALL TOCNT(IDV,N)
      IF(N.EQ.CW)GO TO 7
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
   7  GO TO 3
  11  CALL FMCNT(IDV,M)
      NN=N
      PRINT 4,M,M
      IF (M.EQ.DC1) NN=" "
      GO TO 3
   5  IF (NN.NE."D") THEN
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      ELSE
      WRITE(*,*) " YOU ARE NOT AT THE 1ST CHARACTER OF DATA CHAIN "
      GO TO 12
      ENDIF
      CALL DATAIN8(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL DATAIN8(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      GO TO 3
  12  CALL CLOSECNT(IDV)
      WRITE(*,*) " PROGRAM WILL END IN 2 SEC"
      CALL SLEEP(2)
   2  FORMAT(A1)
   4  FORMAT(1X,'M= ',A1,' Octal :',O16)
   6  FORMAT(1X,I6)
  15  FORMAT(I4)
      STOP
```

```
END
```

onion207.F

```
#define orOPENCNT2071(i,name0) OpenCnt2071(%ref(i),%ref(name0))
C     /* AUTHOR : D. Harvatis - March 88 */
C     /* MODIFIED : G.D. Alexopoulos - April 08*/
C     This subroutine is called by the FORTRAN
C     programs instead of the OpenCnt
C     subroutine.
C     A null character is concatenated to the character
C     argument and the original subroutine is called.
C     However, the character argument that is to be
C     passed to the original routine, must not be
C     longer than 99 characters.
      subroutine OPENCNT2071(i,name)
      character*(*) name
      integer*4 i
      character*100 name0
      l=len_trim(name)+1
      name0=name
      name0(l:l)=char(0)
      call orOPENCNT2071(i,name0)
      return
      end
```

open207dv.c

```
/* AUTHOR : D. Harvatis - March 88 */
/* MODIFIED : G.D. Alexopoulos - April 08*/
#include <errno.h>
#include <unistd.h>
#include <fcntl.h>
#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
extern void SetCnt2071();/*SetCnt2071(dv,t) int dv,t; */
extern void ClearCnt();
struct termios oldset;
void OpenCnt2071(dv,df)
int *dv;
char *df;
{
      if ((*dv = open(df,O_RDWR)) < 0) {
      if (errno==ENAMETOOLONG || errno==ENOTDIR
            ||errno==ENOENT || errno==EISDIR )
            fprintf(stderr,
            "\nOpenCnt : bad file name.\n");
      else if (errno==EMFILE || errno==ENFILE)
            fprintf(stderr,
            "\nOpenCnt : too many open files.\n");
      else if (errno==EACCES || errno==EROFS ||errno==EAGAIN)
            fprintf(stderr,
            "\nOpenCnt : permission denied.\n");
      else
            fprintf(stderr,
            "\nOpenCnt : open failed.\n");
            exit(1);
      }
      /* check if RS-232 is used by another program */
      if (lockf(*dv,F_TLOCK,0) !=0) {
            if (errno==EACCES || errno==EAGAIN)
                  fprintf(stderr,
                  "\nOpenCnt : line unavailable - try again
            later.\n");
            else
                  fprintf(stderr,
                  "\nOpenCnt : unexpected lockf error.\n");
                  exit(1);
      }
      {tcgetattr(*dv,&oldset);
      }
SetCnt2071(*dv,-1);
ClearCnt(*dv);
}
```

mode207dv.c

```
/* AUTHOR : D. Harvatis - March 88 */
/* MODIFIED : G.D. Alexopoulos - April 08*/
#include <errno.h>
#include <termios.h>
#include <stdio.h>
#include <stdlib.h>
void SetCnt2071(dv,t)
int dv,t;
{
      unsigned char c;
      struct termios arg, check;
      if (t != 0)
            c=1;
      else
            c=0;
      arg.c_iflag = IGNPAR;
      arg.c_oflag = 0;
      arg.c_cflag = CS8|CSTOPB|CREAD|HUPCL;
      arg.c_lflag = 0;
      arg.c_cc[VMIN] = c;
      arg.c_cc[VTIME] = 0;
      if(cfsetospeed(&arg,B9600)!=0){
            fprintf(stderr,"\nSetCnt: failed to set port speed.\n");
            exit(1);
      }
      if(tcsetattr(dv,TCSANOW,&arg)==-1){
            fprintf(stderr,"\nSetCnt : tcsetattr failed.\n");
            exit(1);
      }
      tcgetattr(dv,&check);
      if(
      check.c_iflag != arg.c_iflag ||
      check.c_oflag != arg.c_oflag ||
      check.c_lflag != arg.c_lflag ||
      check.c_cc[VMIN] != arg.c_cc[VMIN] ||
      check.c_cc[VTIME] != arg.c_cc[VTIME] ||
      cfgetospeed(&check) != cfgetospeed(&arg)
      ){
      fprintf(stderr,"\nSetCnt: failed to set port parameters.\n");
      exit(1);
      }
}
```

data8dv.c

```
/* AUTHOR : D. Harvatis - March 88 */
/* MODIFIED : G.D. Alexopoulos - April 08*/
#include <stdio.h>
#include <stdlib.h>
#define check(c) (((c)<060)||((c)>071))
extern void FromCnt();
void DataCnt8(dv,err,n)
int dv;
int *err;
long *n;
{
      char s[8];
      int i;
      FromCnt(dv,s);
      if (s[0] == 041) *err = -1;
      else {
            *err=check(s[0]);
            for (i=1; i<8; ++i) {
                  FromCnt(dv,&s[i]);
                  *err += check(s[i]);
            }
            *n = atol(s);
      }
}
```

Σημειώνεται ότι το πρόγραμμα **P20718** που παρουσιάζεται στην εδώ § 5.5.6,μαζί με τα επιπλέον αναγκαία προγράμματα σε C περιέχονται σε CD, που επισυνάπτεται στην παρούσα Διπλωματική Εργασία. Λόγω απλότητας του προγράμματος **P20718**, κρίθηκε ότι δεν είναι απαραίτητο να παρουσιασθεί σε αυτό το σημείο αντίστοιχο λογικό διάγραμμα.

5.5.7 Μετάφραση και σύνδεση

Λαμβάνοντας υπόψιν τα όσα σημειώθηκαν στη παράγραφο 4.8 και δεδομένου ότι το πρόγραμμα **P20718** χρησιμοποιεί για να λειτουργήσει τις υπορουτίνες της βιβλιοθήκης **count**, η μετάφραση και η σύνδεσή του για την παραγωγή του αντίστοιχου εκτελέσιμου **P20718** θα μπορούσε να γίνει από την εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –g *P20718*.F –l count –o *P2071*

η οποία μπορεί να αποτελεί και μέρος άλλου ευρύτερου makefile. Το σχετικό κατάλληλο makefile, μαζί με τα αντίστοιχα κατάλληλα αρχεία count.f.i και counter.h, τροποποιημένα ώστε να περιλαμβάνουν τις σε αυτό το σημείο αναγκαίες αλλαγές στην σειριακή επικοινωνία, θα παρουσιασθούν σε ειδική παράγραφο περί το τέλος του παρόντος Κεφαλαίου.

5.6 Καταμετρητής 2071Aw/02 (Daisy Chain)

Ο καταμετρητής 2071Aw/02 είναι αντίστοιχος με τον 2071A/w03 με τη διαφορά ότι προκειμένου να επικοινωνήσει με άλλους καταμετρητές ή σειριακές θύρες Η/Υ χρησιμοποιεί την τεχνοτροπία Daisy Chain. Για να επικοινωνήσει επομένως με Η/Υ απαιτείται μονάδα επικοινωνίας. Ο καταμετρητής 2071Aw/02 που θα χρησιμοποιηθεί έχει σειριακό αριθμό 186315 και διακρίνεται στις φωτογραφίες των σχημάτων 2.1 (εμπρόσθια όψη) και 2.2 (οπίσθια όψη) του αντίστοιχου εγχειριδίου (βλ. και Παράρτημα). Στη φωτογραφία του Διαγράμματος 5.16, διακρίνεται η πραγματική πίσω όψη του καταμετρητή 2071Aw/02, με την ενσωματωμένες τις 15 pin θύρες για τη σύνδεση Daisy Chain. Σημειώνεται ότι οι ψηφιακοί καταχωρητές του υπόψιν καταμετρητή έχουν μήκος έξι χαρακτήρων. Είναι προφανές ότι για την ανάγνωση αυτού του καταμετρητή από σειριακή θύρα Η/Υ, πρέπει να χρησιμοποιείται μονάδα επικοινωνίας. Από δοκιμές που πραγματοποιήθηκαν στα πλαίσια της παρούσης ΔΕ, διαπιστώθηκε ότι κατάλληλη για το σκοπό αυτό είναι η μονάδα επικοινωνίας 2088. Το πρόγραμμα με το οποίο έγιναν οι σχετικές δοκιμές είναι το **P20887.F**.

5.7 Πρόγραμμα FORTRAN P17887.F

Το πρόγραμμα σε γλώσσα προγραμματισμού FORTRAN77 **P17887.F**, κατασκευάσθηκε για την σύντομη και συνοπτική δοκιμή της πειραματικής διάταξης που περιγράφεται στην παράγραφο 5.3.3. Είναι αντίστοιχο των προγραμμάτων **P20887.F** και **P20718.F** και έχει ως εξής:

P17887.F

```
#include "count.f.i"
      PROGRAM P17887
C     AUTHOR : S.E.SIMOPOULOS
C     HP-UX version by D.HARVATIS  April 88
C     Fedora Core 8 version for 1788 by G.D Alexopoulos May-08
      CHARACTER*1 M,N,KI,K40,K41,K42,K43,KD,KE
      DATA KI,KE,K40,K41,K43,KD/"I","E"," ","!","#","D"/
      CHARACTER*20 TD
      TD="/dev/ttyS0"
      K42=CHAR(34)
      CALL OPENCNT(IDV,TD)
   3  PRINT 1
   1  FORMAT('$ GIVE COMMAND CHARACTER :')
      WRITE(*,*) " R for reset, S for start, H to halt system"
      WRITE(*,*) " A to turn on FP light, B to turn off FP light"
      WRITE(*,*) " D to read 1 char, I to read scalers, E to exit "
      READ(*,2) N
      IF(N.EQ.KI)GO TO 5
      IF(N.EQ.KD)GO TO 11
      IF(N.EQ.KE)GO TO 12
      CALL TOCNT(IDV,N)
      GO TO 3
  11  CALL FMCNT(IDV,M)
      PRINT 4,M,M
      GO TO 3
   5  WRITE (*,*) " PLEASE PRESS FLAG"
      DO WHILE (M.NE.K42)
      CALL FMCNT(IDV,M)
      ENDDO
      CALL TOCNT(IDV,N)
      CALL DATAIN(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      CALL TOCNT(IDV,N)
      CALL DATAIN(IDV,IC,ICNT)
      PRINT 15,IC
      PRINT 6,ICNT
      CALL FMCNT(IDV,M)
      PRINT 4,M,M
      CALL TOCNT(IDV,N)
      GO TO 3
  12  CALL CLOSECNT(IDV)
      WRITE(*,*) " PROGRAM WILL END IN 2 SEC"
      CALL SLEEP(2)
   2  FORMAT(A1)
   4  FORMAT(1X,'M= ',A1,' Octal :',O16)
   6  FORMAT(1X,I6)
```

```
15  FORMAT(I4)
    STOP
    END
```

Λαμβάνοντας υπόψιν τα όσα σημειώθηκαν στη παράγραφο 4.8 και δεδομένου ότι το πρόγραμμα **P20718** χρησιμοποιεί για να λειτουργήσει τις υπορουτίνες της βιβλιοθήκης **count**, η μετάφραση και η σύνδεσή του για την παραγωγή του αντίστοιχου εκτελέσιμου **P20718** θα μπορούσε να γίνει από την εντολή:

$g77 –ffixed-line-length-132 –fno-underscoring –fcase-preserve –g *P17887*.F –l count –o *P17887*

η οποία μπορεί να αποτελεί και μέρος άλλου ευρύτερου makefile.

Το πρόγραμμα **P17887** τρέχει με την απλή εντολή:

./P17887

(αρκεί το αρχείο στο οποίο περιέχεται το προκύπτον εκτελέσιμο να έχει αυτό το όνομα και επίσης να έχει γίνει με ορθό τρόπο η μετάφραση και η σύνδεση).

5.8 Συγκρότηση βιβλιοθήκης και συμπλήρωση makefile

Το σχετικό κατάλληλο makefile, μαζί με τα αντίστοιχα κατάλληλα αρχεία count.f.i και counter.h, τροποποιημένα ώστε να περιλαμβάνουν τόσο τις αναγκαίες αλλαγές και προσθήκες που παρουσιάσθηκαν στο 5^{ο} Κεφάλαιο σε σχέση με το λογισμικό που παρουσιάσθηκε στο προηγούμενο 4^{ο} Κεφάλαιο, όσο και τα αναγκαία προγράμματα σε FORTRAN77 παρουσιάζονται στη συνέχεια.

count.f.i

```
C George Alexopoulos – April 08
#define ITIME(ISEC,IMIN,IHRS,JD,JM,JY) Time_to_I(%ref(ISEC),%ref(IMIN),%ref(IHRS),%ref(JD),%ref(JM),%ref(JY))
#define FMCNT(IDV,M) FromCnt(%val(IDV),%ref(M))
#define TOCNT(IDV,CHI) ToCnt(%val(IDV),%ref(CHI))
#define SETCNT(IDV,T) SetCnt(%val(IDV),%val(T))
#define CLEARCNT(IDV) ClearCnt(%val(IDV))
#define SECNDS(TSTART,TICS,TWAIT,TICS2) Ellapsed(%ref(TSTART),%ref(TICS),%ref(TWAIT),%ref(TICS2))
#define DATAIN(IDV,IC,ICNT) DataCnt(%val(IDV),%ref(IC),%ref(ICNT))
#define CLOSECNT(IDV) CloseCnt(%val(IDV))
#define DATAIN8(IDV,IC,ICNT) DataCnt8(%val(IDV),%ref(IC),%ref(ICNT))
```

makefile

```
#George Alexopoulos & Dimitris Karangelos - April 08
libdir=/usr/local/lib/
count=$(libdir)libcount.a

all: $(count) radon P17887 P20887 P20718 clean

$(count): chardv.o datadv.o opendv.o date.o modedv.o secs.o onion.o open207dv.o mode207dv.o data8dv.o onion207.o

	ar urv $(count) chardv.o datadv.o opendv.o date.o modedv.o secs.o onion.o open207dv.o mode207dv.o data8dv.o
onion207.o
	chmod o+r $(count)

chardv.o: chardv.c counter.h
	gcc -c -g chardv.c -o chardv.o
datadv.o: datadv.c counter.h
	gcc -c -g datadv.c -o datadv.o
data8dv.o: data8dv.c counter.h
	gcc -c -g data8dv.c -o data8dv.o
opendv.o: opendv.c counter.h
	gcc -c -g opendv.c -o opendv.o
open207dv.o: open207dv.c counter.h
	gcc -c -g open207dv.c -o open207dv.o
date.o:   date.c   counter.h
	gcc -c -g date.c -o date.o
modedv.o: modedv.c counter.h
	gcc -c -g modedv.c -o modedv.o
mode207dv.o: mode207dv.c counter.h
	gcc -c -g mode207dv.c -o mode207dv.o
secs.o:   secs.c   counter.h
	gcc -c -g secs.c -o secs.o
onion.o:  onion.F
	g77 -ffixed-line-length-132 -fno-underscoring -fcase-preserve -c -g onion.F -o onion.o
onion207.o:  onion207.F
	g77 -ffixed-line-length-132 -fno-underscoring -fcase-preserve -c -g onion207.F -o onion207.o
```

```
radon: radon.F $(count)
      g77  -ffixed-line-length-132 -fno-underscoring -fcase-preserve -g radon.F -l count -o radon
P17887: P17887.F $(count)
      g77  -ffixed-line-length-132 -fno-underscoring -fcase-preserve -g P17887.F -l count -o P17887
P20887: P20887.F $(count)
      g77  -ffixed-line-length-132 -fno-underscoring -fcase-preserve -g P20887.F -l count -o P20887
P20718: P20718.F $(count)
      g77  -ffixed-line-length-132 -fno-underscoring -fcase-preserve -g P20718.F -l count -o P20718
clean:
      rm -f *.o
```

counter.h

```
/* AUTHOR : D. Harvatis – March 88 */
/* AUTHOR : George Alexopoulos – April 08 */
extern void FromCnt();       /* FromCnt(dv,c)          int dv; char c; */
extern void ToCnt();         /* ToCnt(dv,c)            int dv; char c; */
extern void OpenCnt();       /* OpenCnt(dv,df)         int *dv; char *df; */
extern void OpenCnt2071();   /* OpenCnt2071(dv,df)         int *dv; char *df; */
extern void CloseCnt();      /* CloseCnt(dv)           int dv; */
extern void ClearCnt();      /* ClearCnt(dv)           int dv; */
extern void DataCnt();       /* DataCnt(dv,err,n)      int dv; int *err; long *n; */
extern void DataCnt8();      /* DataCnt8(dv,err,n)      int dv; int *err; long *n; */
extern void SetCnt();        /* SetCnt(dv,t)           int dv,t; */
extern void SetCnt2071();    /* SetCnt2071(dv,t)           int dv,t; */
extern void Time_to_I();     /* Time_to_I(s,m,h,D,M,Y)  int *s,*m,*h,*D,*M,*Y; */
extern void Ellapsed();     /* Ellapsed(sref,tref,s,t) long *sref,*terf,*s,*t; */
```

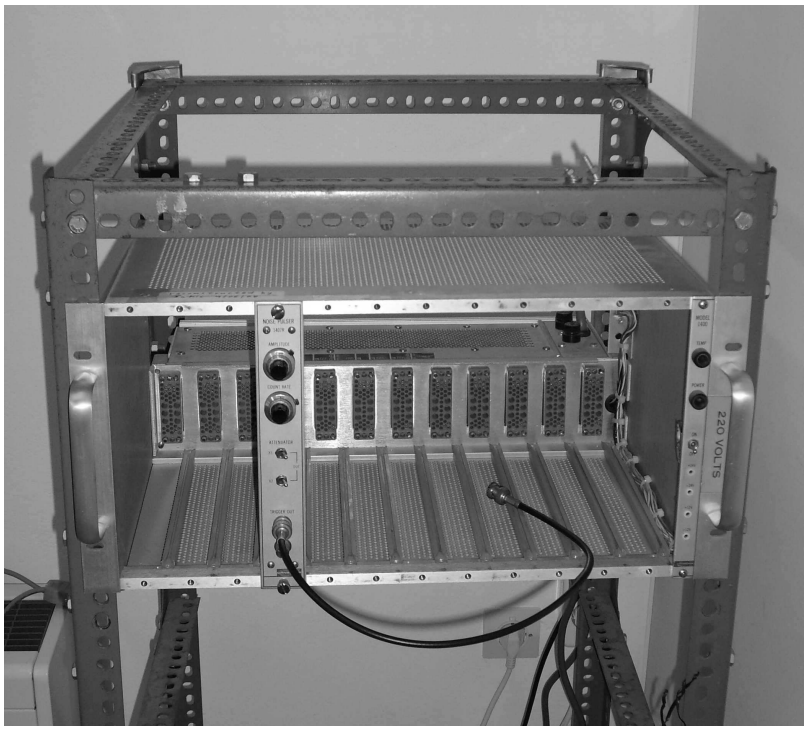
5.9 Συμπεράσματα - Σχόλια

Στις παραγράφους που προηγήθηκαν χρησιμοποιήθηκαν με πολύ επιτυχία προγράμματα εφαρμογής σε γλώσσα προγραμματισμού FORTRAN77 για τον έλεγχο σειριακής πόρτας Η/Υ σε περιβάλλον Λ/Σ LINUX POSIX. Τα προγράμματα αυτά συνεργάσθηκαν απρόσκοπτα με τις αναγκαίες υπορουτίνες σε γλώσσα προγραμματισμού C που παρουσιάσθηκαν στο προηγούμενο 4^{ο} Κεφάλαιο. Στο παρόν 5^{ο} Κεφάλαιο επομένως διαπιστώθηκε η επιτυχής και πλήρης μετατροπή λογισμικού Λ/Σ UNIX SystemV σε λογισμικό Λ/Σ LINUX για τη σύνδεση σειριακών συσκευών μέσω θύρας RS-232. Το λογισμικό που μετατράπηκε δοκιμάσθηκε ώστε να λειτουργεί με όλες τους πιθανούς συνδυασμούς μονάδων σειριακής επικοινωνίας και συνεργαζόμενων καταμετρητών που διαθέτει σήμερα το ΕΠΤ-ΕΜΠ. Πρέπει να τονισθεί στο σημείο αυτό ότι για τον έλεγχο της καλής λειτουργίας του συνόλου του λογισμικού έπαιξε πολύ σημαντικό ρόλο και η αποδοτική λειτουργία του κυκλώματος "υποκλοπής" σειριακών επικοινωνιών, όπως αυτό περιγράφθηκε στο 3^{ο} Κεφάλαιο.

ΔΙΑΓΡΑΜΜΑΤΑ ΤΟΥ 5^{ου} ΚΕΦΑΛΑΙΟΥ

Διάγραμμα 5.1

Μονάδα NIM Bin σε μικρό φορητό ικρίωμα



Διάγραμμα 5.2

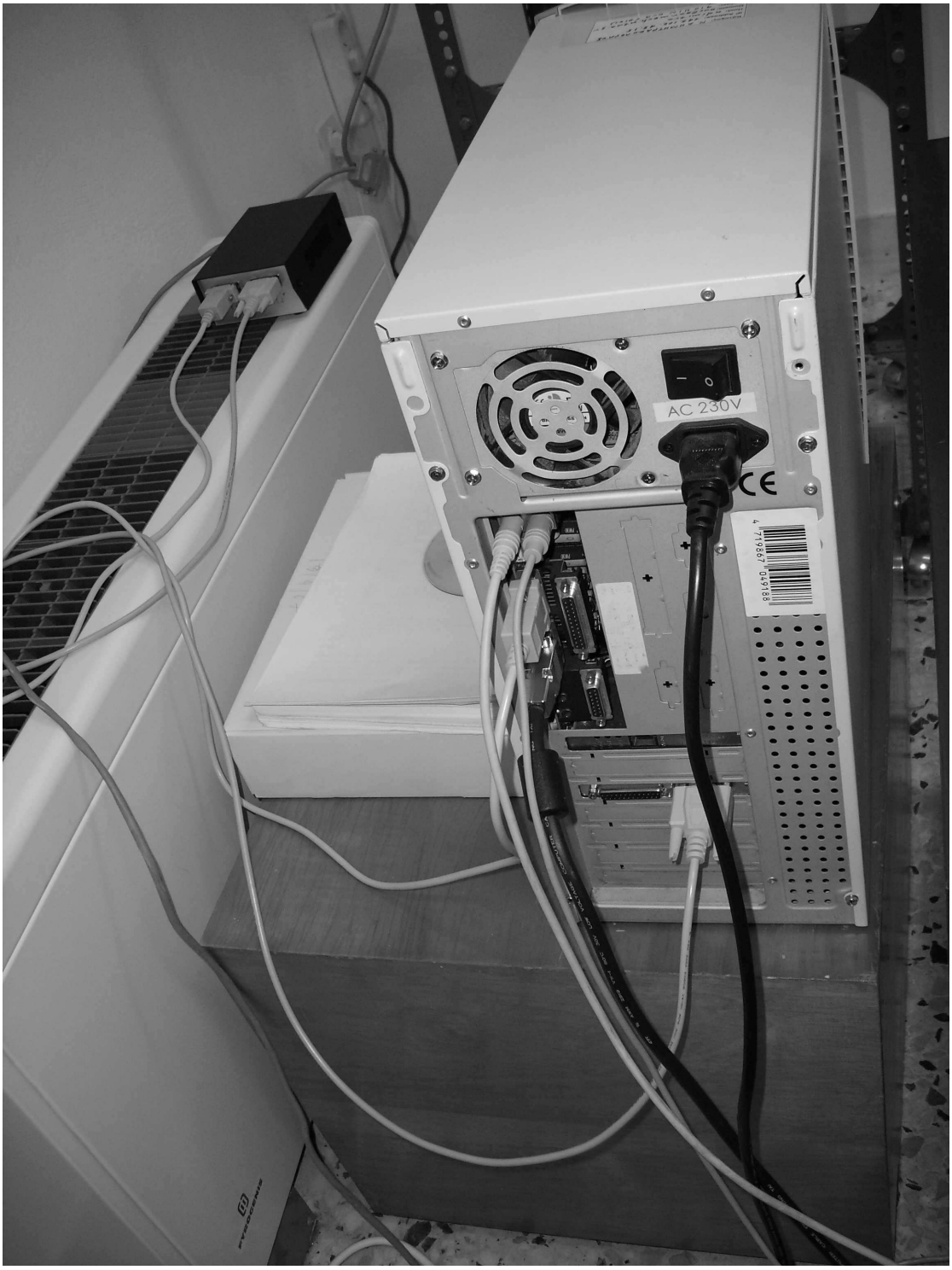
Παλμοδοότης CANBERRA Model 1407R, τοποθετημένος στη NIM Bin

Διάγραμμα 5.3

Συγκροτημένη πειραματική διάταξη.
Από αριστερά: Υπολογιστής με Λ/Σ LINUX, Υπολογιστής "υποκλοπής" σειριακού σήματος και rig για καταμετρητές και μονάδες σειριακής επικοινωνίας.

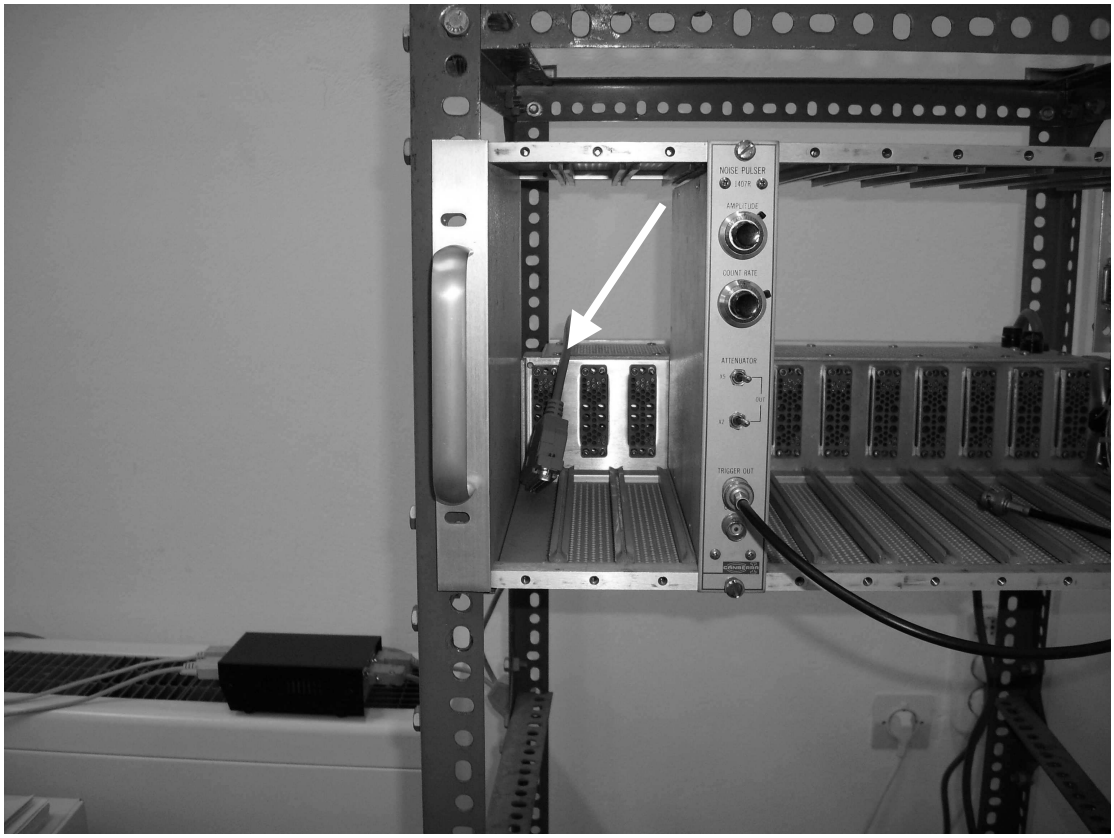
Διάγραμμα 5.4

Διαδρομή σειριακή σύνδεσης από τον βασικό Η/Υ (αριστερά, λευκό τοξάκι) προς το ηλεκτρονικό κύκλωμα "υποκλοπής" σειριακού σήματος (δεξιά, λευκό τοξάκι) και τελικά προς την ελεγχόμενη διάταξη (μαύρο τοξάκι)



Διάγραμμα 5.5

Διαδρομή σειριακής σύνδεσης από το ηλεκτρονικό κύκλωμα "υποκλοπής" σειριακού σήματος (αριστερά) προς δύο σειριακές θύρες του Η/Υ "υποκλοπής"



Διάγραμμα 5.6

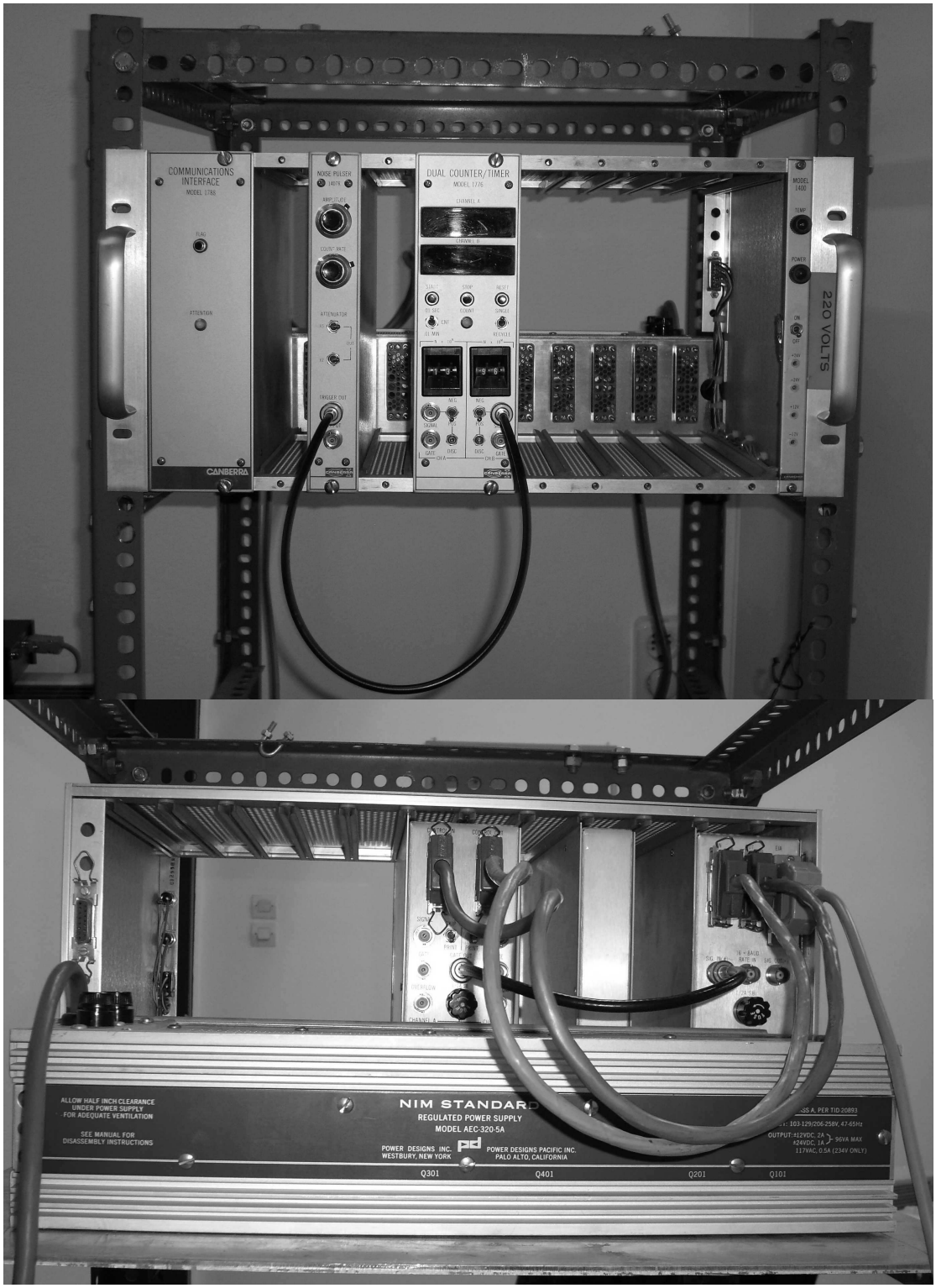
Αρσενικό βύσμα DB25, σε αναμονή (λευκό τοξάκι) κατάλληλο για να συνδεθεί στην θηλυκή υποδοχή DB25 της σειριακής συσκευής με την οποία χρειάζεται να αποκατασταθεί επικοινωνία.

Διάγραμμα 5.7

Εμπρόσθια όψη μονάδας επικοινωνίας 1788.

Διάγραμμα 5.8

Πίσω όψη μονάδας επικοινωνίας 1788.

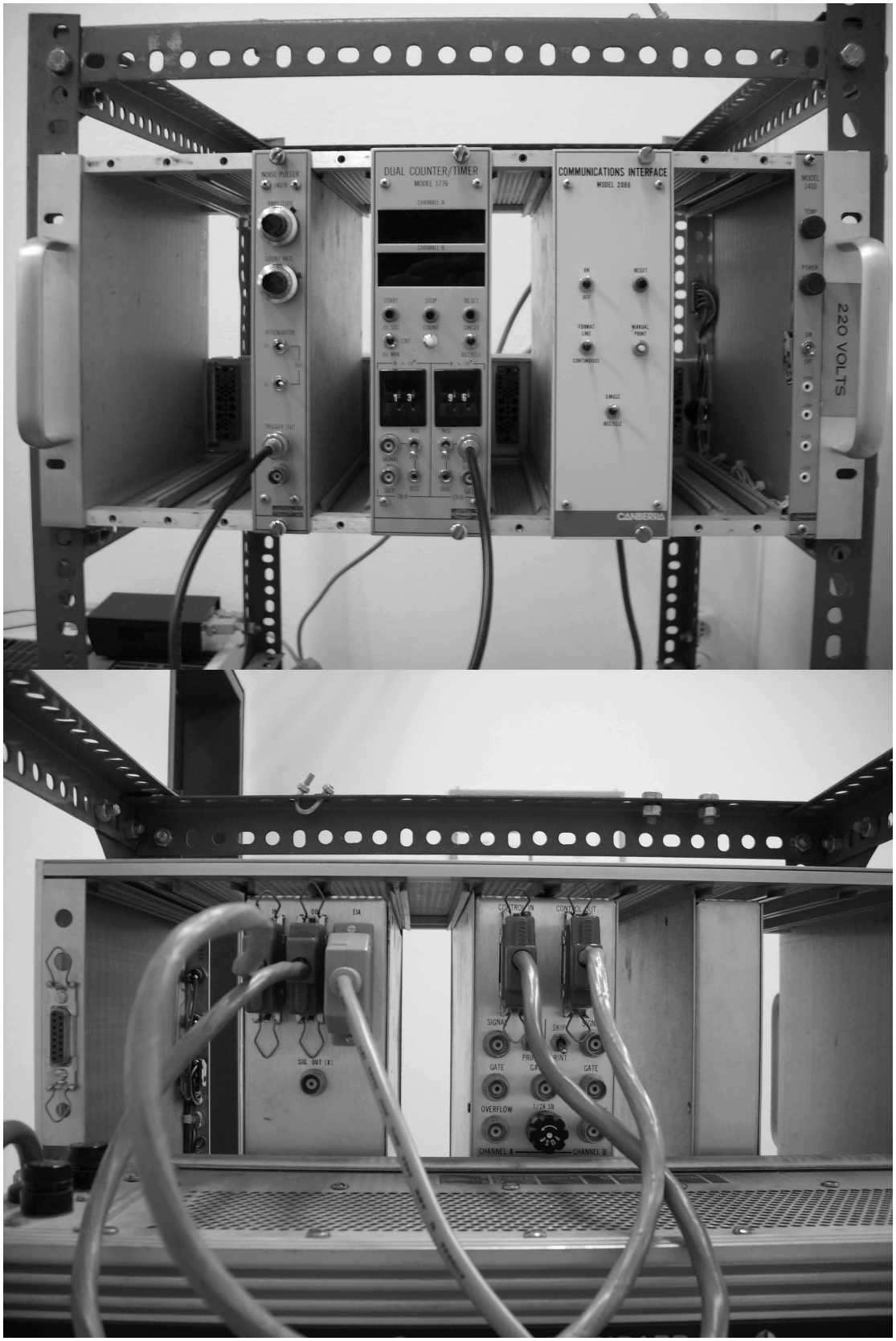


Διάγραμμα 5.9

Ο καταμετρητής 1776 συνδεδεμένος με τη μονάδα επικοινωνίας 1788 με καλωδίωση Daisy Chain.

Διάγραμμα 5.10

Σειριακή σύνδεση μονάδας επικοινωνίας 1788.

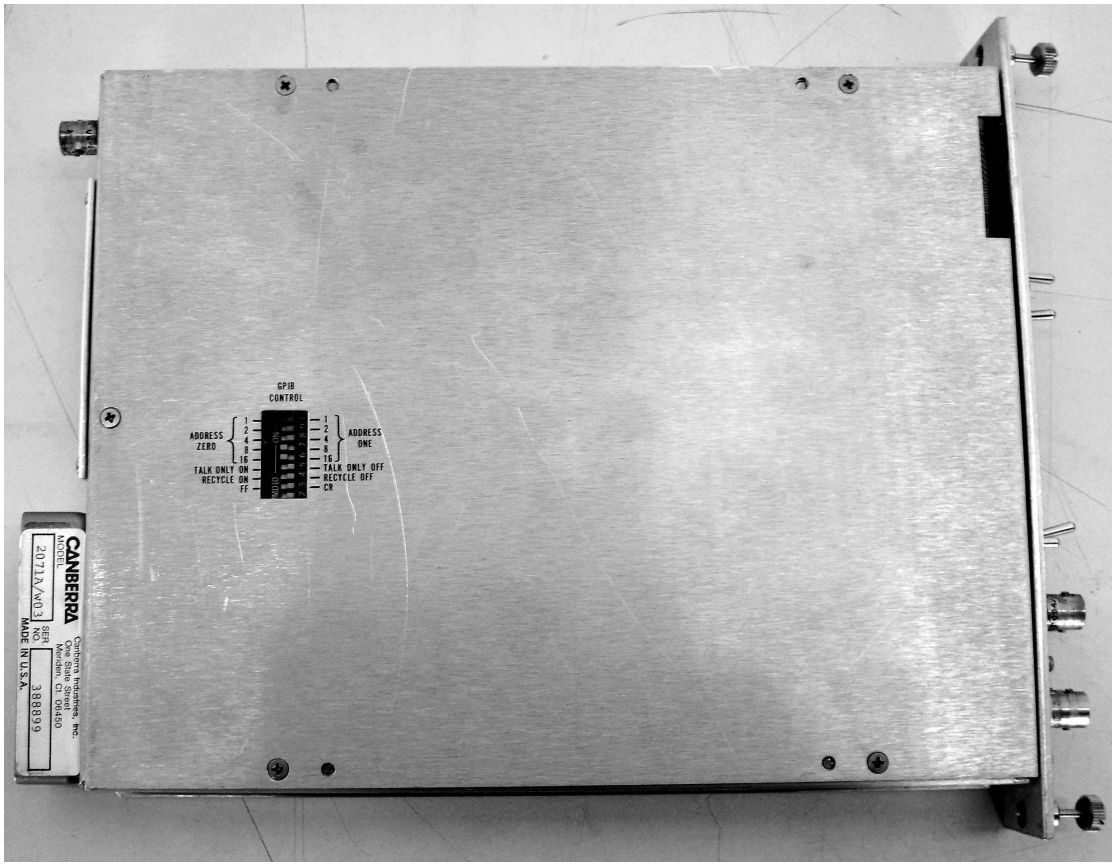


Διάγραμμα 5.11

Ο καταμετρητής 1776 συνδεδεμένος με τη μονάδα επικοινωνίας 2088 με καλωδίωση Daisy Chain.

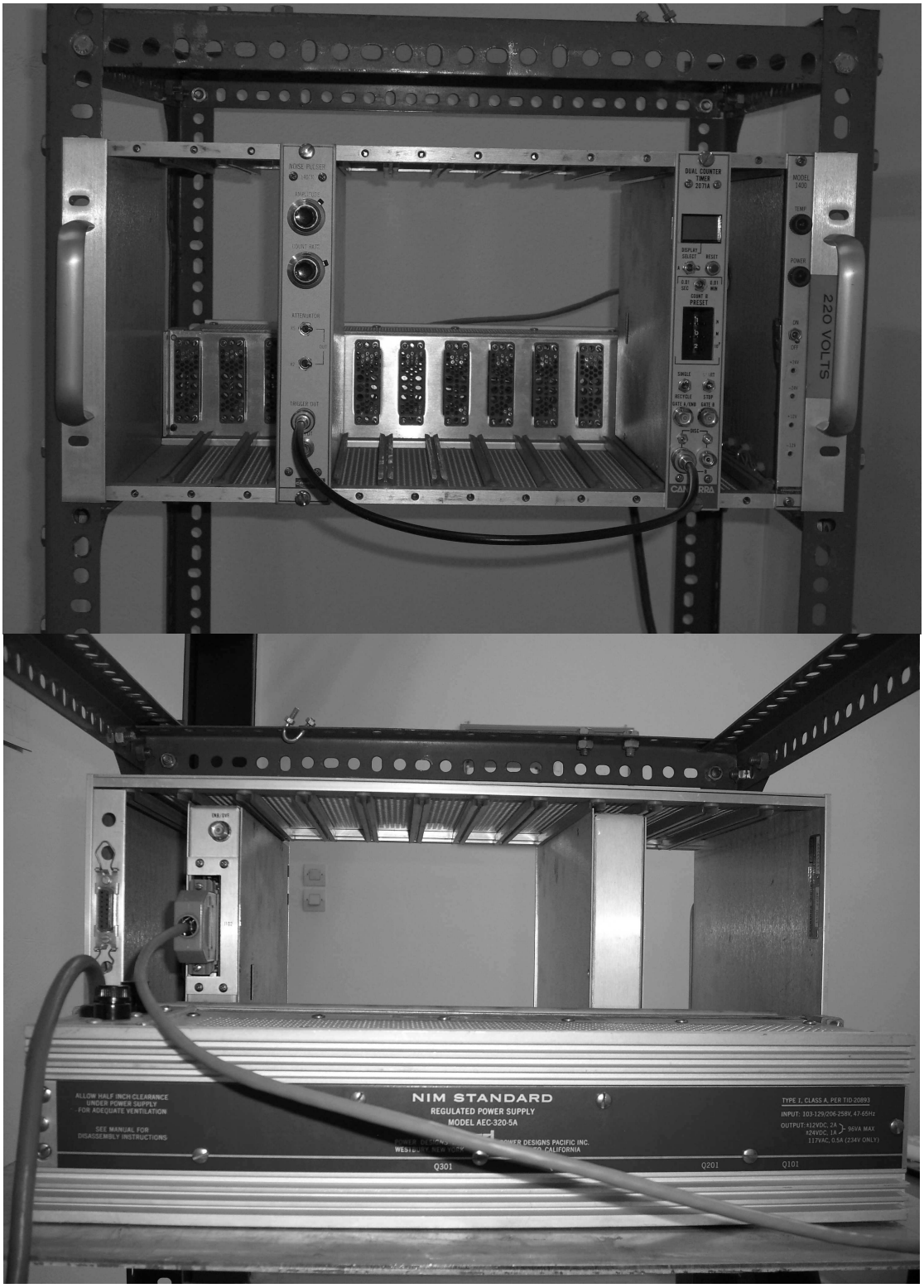
Διάγραμμα 5.12

Ο καταμετρητής 2071A/w03 (πίσω όψη).



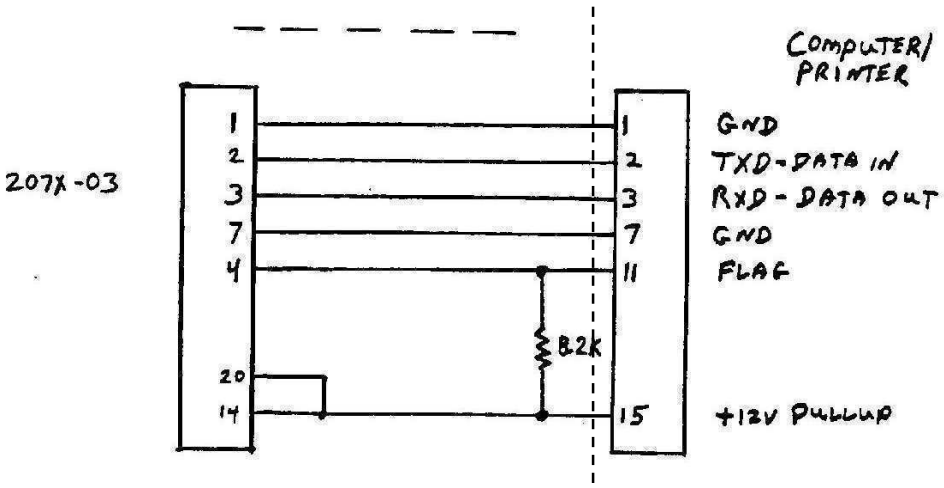
Διάγραμμα 5.13

Ο καταμετρητής 2071A/w03 (πλάγια όψη).



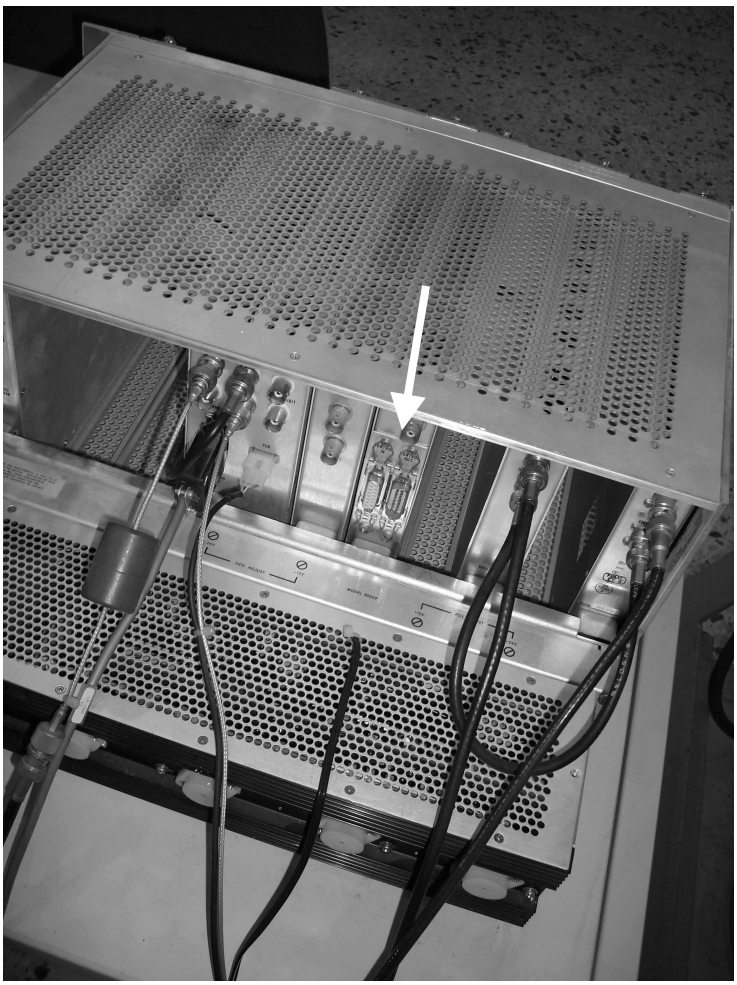
Διάγραμμα 5.14

Ο καταμετρητής 2071A/w03 με το σειριακό του καλώδιο



Διάγραμμα 5.15

Το σειριακό του καλώδιο του καταμετρητή 2071A/w03 (αριστερά από τη διακεκομμένη γραμμή).



Διάγραμμα 5.16

Η πίσω όψη του καταμετρητή 2071Aw/02.

ΕΠΙΛΟΓΟΣ

Σύμφωνα με τα όσα παρουσιάσθηκαν και αναλύθηκαν στα προηγούμενα Κεφάλαια στα πλαίσια της παρούσης ΔΕ και κυρίως σύμφωνα με όσα σημειώθηκαν στο 4^{ο} και το 5^{ο} Κεφάλαιο προέκυψαν συμπεράσματα, τα οποία μπορούν να συνοψισθούν ως εξής:

(α) Τα προγράμματα σε γλώσσα προγραμματισμού C, καθώς και εκείνα σε γλώσσα προγραμματισμού FORTRAN, που αναπτύχθηκαν από τον Χαρβάτη (1990) για τον έλεγχο σειριακής πόρτας Η/Υ σε περιβάλλον Λ/Σ UNIX System V, είναι γενικά συμβατά με περιβάλλον Λ/Σ LINUX POSIX. Στην υπολογιστική γλώσσα μπορεί δηλαδή να γράφεται ότι διαπιστώθηκε portability (στα αδόκιμα ελληνικά "φορητότητα").

(β) Οι αλλαγές και οι μεταβολές που χρειάσθηκαν να γίνουν στα υπόψιν πιοπάνω αναφερόμενα προγράμματα ήταν γενικά μικρές, εύκολες και κατανοητές. Με τις αλλαγές επιτεύχθηκε ο στόχος της ΔΕ που και κατά τον τίτλο προσδιορίζεται ως η "ΜΕΤΑΤΡΟΠΗ ΛΟΓΙΣΜΙΚΟΥ Λ/Σ UNIX SYSTEM V ΣΕ ΛΟΓΙΣΜΙΚΟ Λ/Σ LINUX ΓΙΑ ΤΗ ΣΥΝΔΕΣΗ ΣΕΙΡΙΑΚΩΝ ΣΥΣΚΕΥΩΝ ΜΕΣΩ ΘΥΡΑΣ RS-232"

(γ) Το λογισμικό που μετατράπηκε δοκιμάσθηκε ώστε να λειτουργεί με όλες τους πιθανούς συνδυασμούς μονάδων σειριακής επικοινωνίας και συνεργαζόμενων καταμετρητών που διαθέτει σήμερα το ΕΠΤ-ΕΜΠ. Εξαιρούνται οι καταμετρητές CANBERRA Model 512, οι οποίοι είναι νεότερο απόκτημα στο ΕΠΤ-ΕΜΠ.

(δ) Για τον έλεγχο της καλής λειτουργίας του συνόλου του λογισμικού έπαιξε πολύ σημαντικό ρόλο και η αποδοτική λειτουργία του κυκλώματος "υποκλοπής" σειριακών επικοινωνιών, όπως αυτό περιγράφθηκε στο 3^{ο} Κεφάλαιο.

και

(ε) Το κείμενο της παρούσας ΔΕ, συγκροτήθηκε με τρόπο ώστε όχι μόνο να παρουσιάζει το λογισμικό που μετατράπηκε αλλά και να τεκμηριώνει με επαρκή λεπτομέρεια τον τρόπο που αυτό χρησιμοποιείται ή μπορεί να χρησιμοποιηθεί στο ΕΠΤ-ΕΜΠ. Για το σκοπό αυτό χρησιμοποιήθηκαν πηγές

οι κυριότερες από τις οποίες (εκτός από τα αντίστοιχα εγχειρίδια των συσκευών) είναι: 1. Σιμόπουλος (1986), 2. Δ.Ε. Χαρβάτη (1990), 3. Δ.Ε. Βοσυνιώτη (1990), 4. Σιμόπουλος (1994 και 1997).

Συμπληρωματικά με τα παραπάνω αξίζει να αναφερθεί ότι:

- Αντί μιας από τις τυποποιημένες σειριακές θύρες RS-232 ενός PC συμβατού Η/Υ (π.χ. της /dev/ttyS0), μπορεί να χρησιμοποιηθεί με την ίδια αποτελεσματικότητα μία USB σειριακή θύρα (π.χ. η /dev/ttyUSB0). Προυποτίθεται ότι διατίθεται ο κατάλληλος για το σκοπό αυτό μετατροπέας USB – to – serial adapter ή converter. Στο ΕΠΤ-ΕΜΠ μετά από πολλές δοκιμές διαφόρων τέτοιων μετατροπέων της αγοράς διαπιστώθηκε (Καράγγελος, 2007) ότι οι πλέον κατάλληλοι τέτοιοι είναι αυτοί που δηλώνονται ως "modem capable" ή ως "Suitable for PDA and Modem". Ενας τέτοιος μετατροπέας είναι ο τύπος DA-70146-VPR 3.0 της εταιρείας κατασκευής Digitus (FTDI), σε τυποποίηση USB 2.0, ο οποίος είναι διαθέσιμος σήμερα (2008) στην Ελληνική αγορά και από την εταιρία COSMODATA (http://www.cosmodata.gr). Με τη χρήση USB – to – serial adapter, είναι δυνατόν να χρησιμοποιηθεί και να λειτουργήσει το λογισμικό που αναπτύχθηκε και από φορητούς Η/Υ, οι οποίοι συνήθως (πλην πολύ ακριβών εξαιρέσεων) σήμερα (2008) δεν διαθέτουν πλέον σειριακές θύρες RS-232. Σε σχετικές επιτυχημένες δοκιμές που πραγματοποιήθηκαν διαπιστώθηκε ότι στην ατυχή περίπτωση που το λογισμικό χρησιμοποιηθεί εσφαλμένα (π.χ. χωρίς να υπάρχει σχετική μονάδα επικοινωνίας πίσω από την αντίστοιχη θύρα RS-232 του Η/Υ), η σειριακή θύρα π.χ. /dev/ttyUSB0, δεν "κολλάει" (στα αγγλικά hung).

- Στην ατυχή περίπτωση που η χρησιμοποιούμενη σειριακή θύρα (π.χ. η /dev/ttyS0) "κολλήσει", μπορούν να χρησιμοποιηθούν διάφοροι τρόποι για επανεκκίνηση της θύρας. Αυτός που προτιμάται εδώ, είναι να δοθεί η εντολή από root:

  # agetty –L 9600 /dev/ttyS0 vt100

  ακολουθούμενη από **CTL-C**

Πέρα από τα πιοπάνω συμπεράσματα και τις συμπληρωματικές παρατηρήσεις, προκύπτουν οι ακόλουθες προτάσεις για μελλοντική Διπλωματική Εργασία:

- Κυρίως σε ότι αφορά το πρόγραμμα RADON ($5^{ο}$ Κεφάλαιο), μπορεί να παρατηρηθεί, ότι λόγω διαδοχικών μετατροπών του για διάφορα Λ/Σ και διάφορες εκδόσεις γλώσσας FORTRAN υστερεί σε χαρακτηριστικά δομημένου προγραμματισμού. Το πρόγραμμα αυτό θα μπορούσε να ξαναγραφεί στα πλαίσια επόμενης Δ.Ε.

- Σε σχέση με τους καταμετρητές CANBERRA Model 512, από τους οποίους το ΕΠΤ-ΕΜΠ διαθέτει ήδη δύο και οι οποίοι έχουν τη δυνατότητα σειριακής σύνδεσης, σε επόμενη Δ.Ε. μπορεί να παραγχθεί κατάλληλο λογισμικό ώστε και αυτές οι συσκευές να αρχίσουν να χρησιμοποιούνται παραγωγικά.

Τέλος:

Ως αναπόσπαστο μέρος της παρούσης Δ.Ε. επισυνάπτεται (α) CD, στο οποίο περιέχεται όλο το σχετικό λογισμικό που χρησιμοποιήθηκε και (β) DVD με το Λ/Σ Linux έκδοση Fedora Core 8 (Werewolf). Όπως έχει ήδη σημειωθεί το CD περιέχει τόσο τα αναγκαία προγράμματα εγκατάστασης της γλώσσας FORTRAN που χρησιμοποιείται, όσο και το λογισμικό υποκλοπής σειριακής επικοινωνίας για Λ/Σ MS Windows 98.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Anagnostakis M.J., Hinis E.P., Simopoulos S.E., Angelopoulos M.G., "Natural Radioactivity Mapping of Greek Surface Soils", Environment International 22 (SUPPL. 1), S3-S8, 1997
2. Βοσυνιώτης Σ., "Σύνδεση Ψηφιακών Συσκευών σε Θύρα RS-232 Η/Υ κάτω από το Λ/Σ DOS", Διπλωματική Εργασία, ΕΜΠ Τομέας Πυρηνικής Τεχνολογίας, Αθήνα, 1990.
3. CANBERRA http://www.canberra.com/pdf/Products/NIM_pdf/Model_512_super_spec.pdf)
4. CANBERRA Product Catalog, Edition Twelve, Canberra Industries, Meriden, Connecticut, USA
5. CANBERRA Product Catalog, Edition Ten, Canberra Industries, Meriden, Connecticut, USA
6. CANBERRA Instruments Catalog, Edition Nine, Canberra Industries, Meriden, Connecticut, USA
7. CANBERRA, Edition Eight, Canberra Nuclear Products Group, Meriden, Connecticut, USA
8. CANBERRA, Edition Seven, Canberra Industries, Meriden, Connecticut, USA
9. CANBERRA, Edition Six, Canberra Industries, Meriden, Connecticut, USA
10. CANBERRA 81-82, Canberra Industries, Meriden, Connecticut, USA
11. CANBERRA 79-80, Canberra Industries, Meriden, Connecticut, USA
12. CANBERRA 77, Canberra Industries, Meriden, Connecticut, USA, 1977
13. CANBERRA 76, Canberra Industries, Meriden, Connecticut, USA, 1976
14. Fujitaka K., Matsumoto M., Kaiho K., Abe S., "Effect of Rain Interval on Wet Deposition of Radon Daughters", Radiation Protection Dosimetry, 45(1/4): 333-336, 1992
15. Hewlett Packard, "FORTRAN / 9000 Reference", HP 9000 Series 300 Computers, HP Part Number 97081-90041 & 97081-90802, 1987
16. Hewlett Packard, "HP-UX Reference", vol.3: Sections 2,3,4,5, and 7, HP 9000 Series 300 Computers, HP-UX Release 6.2, HP Part Number 09000-90009, Edition 4, 1988

17. Instruction Manual 1788, "Canberra Asynchronous Line Interface for C.I. Data System Modules, Model 1788", Canberra,1978
18. Karangelos D.J., Petropoulos N.P., Anagnostakis M.J., Hinis E.P., Simopoulos S.E., "Data Leading to the Investigation of a Relation between Seismic Activity and Airborne Radon Decay Product Concentrations Outdoors", The Natural Radiation Environment VII (Proceedings), Radioactivity in the Environment Series, Volume 7: 187-197, Elsevier, 2005.
19. Καράγγελος Δ.Ι, Προσωπική Επικοινωνία, 2007
20. Καράγγελος Δ.Ι., Σιμόπουλος Σ.Ε., Προσωπική Επικοινωνία, 2008
21. Λεωνίδου Δ.Ι., "Μέθοδοι και Όργανα Ανιχνεύσεως Πυρηνικών Ακτινοβολιών", ΕΜΠ Τομέας Πυρηνικής Τεχνολογίας, Αθήνα 1986.
22. Μητρόπουλος Η., "Σύνδεση Συσκευών Απευθείας στο Bus Η/Υ κάτω από το Λ/Σ UNIX", Διπλωματική Εργασία, ΕΜΠ Τομέας Πυρηνικής Τεχνολογίας, Αθήνα, 1994.
23. Operator's Manual 2088 "Communications Interface, Model 2088", Canberra, 1987
24. Operator's Manual 207x-03 "EIA Interface, Model 207x-03", Canberra, 1988
25. Πετρόπουλος Α., Προσωπική Επικοινωνία, 2008
26. Simopoulos S.E., "Soil sampling and ^{137}Cs analysis of the Chernobyl fallout in Greece", Applied Radiation and Isotopes, 40(7): 609-613, 1989.
27. Σιμόπουλος Σ.Ε., "Προγράμματα για την 'απευθείας' σύνδεση (on line) των Η/Υ του Εργαστηρίου Πυρηνικής Τεχνολογίας ΕΜΠ", ΕΜΠ Τομέας Πυρηνικής Τεχνολογίας, Αθήνα 1986.
28. Σιμόπουλος Σ.Ε., Προσωπικές Σημειώσεις, 1994
29. Σιμόπουλος Σ.Ε., Προσωπικές Σημειώσεις, 1997
30. Χαρβάτης Δ., "Σύνδεση Ψηφιακών Συσκευών σε Θύρα RS-232 Η/Υ κάτω από το Λ/Σ UNIX", Διπλωματική Εργασία, ΕΜΠ Τομέας Πυρηνικής Τεχνολογίας, Αθήνα, 1990.
31. Χίνης Ε.Π., Σιμόπουλος Σ.Ε., Προσωπική Επικοινωνία, 1995

ПАРАРТНМА

Model 1772 Counter/Timer

Features

- Timer and scaler operation
- Solid state LED display (Light-Emitting Diodes)
- Built-in integral discriminator
- Electronic gating capability
- Accepts positive or negative logic input
- Pulse pair resolution less than 40 nanoseconds
- Seven decade count capacity with built-in prescaler

Description

The Canberra Model 1772 Counter/Timer combines in a dual-width NIM module one six-decade (crystal optional) 25MHz non-printing scaler slaved to an internal line frequency timer. The module may function as a system master to control the duration of counting experiments on the basis or preset time utilizing electronic gating techniques.

Independent thumbwheel switches select the preset time limit for the switch selectable time base of 0.1 seconds or 0.01 minutes. The N switch selects a multiplier (0-9) and the 10^M switch selects the power of ten (10^0 - 10^5) by which it is multiplied.

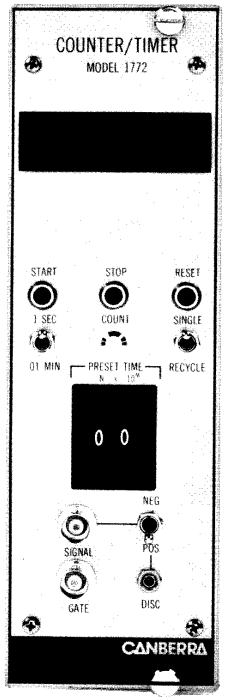
The scaler input is equipped with an integral discriminator covering an input voltage range of +0.5 to +10 volts by means of a front panel screwdriver adjustment. This discriminator can replace a separate integral discriminator in many counting systems.

The six numeric indicators utilize seven-segment light-emitting diodes to provide solid state reliability and long life. Because this is a flat surface display, the viewing angle is far wider than that obtained with tube display. A circularly-polarized filter allows brilliant readout even in direct sunlight. Leading zeros are suppressed for viewing ease.

The counter/timer may operate in Single (one cycle) or Recycle modes. In single mode, counting stops when the preset time is reached and the unit will await new manual commands. In Recycle mode, counting stops momentarily for ≈2 seconds, then the timer and scaler are reset and counting begins again.

The Model 1772 Counter/Timer features fast gating capability. An internal jumper allows the gate input to control only the scaler or both the timer and the scaler. The gate output is controlled by the preset time or by the stop pushbutton and is activated until preset is reached or the stop pushbutton is depressed, except when inhibited by the gate input signal. A front panel indicator lights when the scaler is in a condition to accept counts.

Front panel Start, Stop and Reset pushbutton switches are provided for manual control of the nuclear counter



Specifications

INPUTS

POSITIVE SIGNAL IN - accepts 0.5 to 10 volts (+50 volt maximum) pulses; any rise time; minimum pulse width, 20 nanoseconds; input impedance greater than 1000 ohms; DC coupled; front and rear panel BNC connectors

NEGATIVE SIGNAL IN - accepts -0.6 to -2.0 volt pulses; minimum width, 20 nanoseconds; input impedance 50 ohms; DC coupled; front and rear panel BNC connectors

GATE IN - +2 to +10 volt DC level or pulses (or open circuit) enables accumulation; less than 0.8 volt DC level or pulse (or short circuit) inhibits accumulation; 3mA current sinking; input impedance greater than 1000 ohms, DC coupled, front and rear panel BNC connectors

OUTPUTS

OVERFLOW - +5 volt pulses at reset of sixth decade; rise time less than 200 nanoseconds, width 5 microseconds; DC coupled; rear panel BNC connector

GATE OUT - provides +5 volt DC level during accumulation, +0.5 volt DC level when not accumulating (except when inhibited by Gate Input); rise time less than 200 nanoseconds; will gate up to 50 compatible scalers and timers; current sinking 150MA, DC coupled; rear panel BNC connector

FRONT PANEL CONTROLS

START - pushbutton to renew or begin accumulation

STOP - pushbutton to terminate accumulation

Specifications(continued)

RESET - pushbutton to reset all decades to zero; accumulation will begin unless the Stop pushbutton is first depressed

MODE - toggle switch to select Single or Recycle mode of operation upon reaching a preset condition

PRESET - two digit thumbwheel switches for the time base; selectable preset limit is N x 10^M, N is the 0-9 digit multiplier and 10^M is the power of ten being multiplied; N = 0 is preset OFF, disabling the preset functions of the time base

NEG./POS. - front panel toggle switch to select input pulse polarity for the scaler

DISC. - front panel screwdriver adjustment to inhibit counting of input pulses to the scaler below selected level from +0.5 to 10 volts

INTERNAL CONTROLS

PRESCALER - jumper plugs to insert a decade prescaler in the desired channel; this prescaler is non-displaying

GATE MODE - jumper plug allows the gate input to control either the scaler or both the timer and the scaler

PERFORMANCE

CAPACITY - six decades; 10^6-1 counts; a non-displaying decade may be selected by internal jumper

COUNT RATE - 25MHz minimum, 30MHz typical

PULSE PAIR RESOLVING TIME - less than 40 nanoseconds

TIME BASE ACCURACY - ≈0.1% (or if the crystal controlled option is present, 0.007%) line frequency accuracy

TEMPERATURE OPERATING RANGE - 0 to 50°C

AUTOMATIC RESET - initiated when power is first turned on or after power failed; resets all decade counters to zero

INDICATORS

VISUAL DISPLAY - six solid state numeric indicators for each channel; leading zeros are suppressed

COUNT - front panel light is illuminated during accumulation and extinguished when Stop command is initiated

CONNECTORS

SIGNAL IN, GATE IN - front and rear, BNC, UG-1094/U

OVERFLOW, GATE OUT - rear panel, BNC, UG-1094/U

POWER

+24V	0mA	+12V	50mA
-24V	0mA	-12V	150mA
115VAC	200mA		

PHYSICAL

SIZE - standard double width module (2.70 inches wide) (6.85 cm wide) per TID-20893

NET WEIGHT - 3.2 lbs. (1.47 kgs)

SHIPPING WEIGHT - 8.2 lbs. (3.69 kg)

Model 1776
Dual Counter/Timer

Features

- Dual scaler or timer and scaler operation
- Dual solid state LED display (Light-Emitting Diode)
- Preset time and/or count operation
- Built-in integral discriminators
- Electronic gating capability
- Positive or negative logic input for each scaler
- Printout capability with standard Canberra NIM modules
- Pulse pair resolution less than 40 nanoseconds
- Seven decade count capacity with built-in prescalers

Description

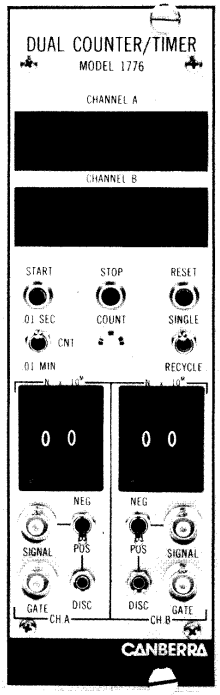
The Canberra Model 1776 Dual Counter/Timer combines in a dual-width NIM module two independent six decade scalers, plus an internal crystal controlled time base. Two presettable counting channels, identified as A and B permit preset count and preset time operation either separately or simultaneously. The module may function as a system master to control the duration of counting experiments on the basis of preset count and/or time, or may serve simply as a dual channel system slave, operating at counting rates up to 25MHz.

Independent thumbwheel switches select the preset count or time limit for each channel. The N switch selects a Multiplier (0-9) and the 10^M switch selects a power of ten (10^0-10^5) by which it is multiplied. Other features include: front panel independent electronic gating; integral discrimination (+0.5 to +10 volt range) of both channel A and B input signals; manual start, stop, and reset controls; single and recycle modes of operation; positive and negative input pulse selection on each channel; and a rear panel GATE OUT connector for gating up to 50 compatible scalers and timers.

Accumulated data printout of the two internal scalers is accomplished with standard Canberra Print/Control NIM modules such as the Model 1488 Teletype Scanner. Printout of either channel may be inhibited by rear panel switch. When employed in a printing system, the printout controller generally establishes single or recycle mode of operation. In a non-printing system, the Model 1776 may be operated in either the single or recycle mode. In the recycle mode the display time between recycles is internally adjustable from 1 to 5 seconds.

The Model 1776 display utilizes light-emitting diodes to provide solid state reliability and long life. Because this is a flat-surface display, the viewing angle is far wider than that obtained with tube type displays. A circularly polarized filter allows brilliant readout even in direct sunlight. Two sets of six numeric indicators display the contents of Channel A and B independently.

Channel A has the capability of presetting and counting either the internal time base (0.01 sec. or 0.01 min.) or the external A count input signals (CNT), depending upon the setting of the front panel channel A switch labeled .01 min/CNT/.01 sec. Channel B is a second scaler which is presettable and independent of channel A. The B scaler counts and displays the B count input signals.



Specifications

INPUTS

SIGNAL

POS. - channel A and B; accept +0.5 to +10 volts (+50 volt maximum) pulses; any rise time; minimum pulse width, 20 nanoseconds; input impedance greater than 1000 ohms; DC coupled; front and rear panel BNC connectors

NEG. - channel A and B; accept -0.6 to -2 volt pulses; 20 nanoseconds minimum pulse width; input impedance 50 ohms; DC coupled; front and rear panel connectors

GATE - channel A and B; +3 to +10 volt DC level or pulses (or open circuit) enables accumulation; less than 0.8 volt DC level or pulse (or short circuit) inhibits accumulation; input impedance greater than 1000 ohms, DC coupled, front and rear panel connectors

CONTROL IN - accepts START, STOP, RESET, PRINT (5 volt negative-going pulses from +5 volt level, rise time 500 nanoseconds maximum); width 1.0 microsecond minimum, and HOLD commands (DC level change from +5 volts to +0.5 volts maximum during printout); rear panel Amphenol 17-10150 15-pin connector

OUTPUTS

OVERFLOW - +5 volt pulses at reset of sixth decade; rise time less than 250 nanoseconds, width 0.5 microseconds; DC coupled; rear panel BNC connector

GATE OUT - provides +5 volt DC level during accumulation, +0.5 volt DC level when not accumulating (except when inhibited by Gate Input); rise time less than 25 nanoseconds; will gate up to 50 compatible scalers and timers; current sinking 150mA, DC coupled; rear panel BNC connector

Specifications(continued)

CONTROL OUT - provides START, STOP, and RESET commands (5 volt negative-going pulses from +5 volt level, rise time 500 nanoseconds maximum, width 1.0 microsecond minimum); data output information in the form of decimal digit serial 1-2-4-8 BCD; logic 1 = +4 volts DC, logic 0 = 0 volts DC; connector also provides "next unit" PRINT command; eighth and successive Print command pulses are routed out via logic gate; all control lines except Print command are wired directly between Control Input-Control Output connectors; rear panel Amphenol 17-20150 15 pin connector

FRONT PANEL CONTROLS

START - manual pushbutton to renew or begin accumulation

STOP - manual pushbutton to terminate accumulation

RESET - manual pushbutton to reset all decades to zero; accumulation will begin unless the stop pushbutton is first depressed

CHANNEL A TIME BASE or COUNT - three position toggle switch to select 0.01 minute or 0.01 second time base; CNT position permits channel A to function as a presettable counter

MODE - toggle switch to select Single or Recycle mode of operation upon reaching a preset condition

PRESET - two digit thumbwheel switches for each channel; selectable preset limit is N x 10^M, N is the 0-9 digit multiplier and 10^M is the power of ten being multiplied; either channel N = 0 is preset OFF, disabling the preset functions of that channel

NEG./POS. - two front panel toggle switches to select input pulse polarity for channels A and B

DISC. - two front panel screwdriver, adjustments to inhibit counting of input pulses to channels A and B below selected level from +0.5 to 10 volts

REAR PANEL CONTROLS

PRINT - two rear panel toggle switches (channels A and B) to enable (PRINT) or inhibit (SKIP) printout of either or both of the two scalers

INTERNAL CONTROLS

PRESCALER - jumper plugs to insert a decade prescaler in the desired channel; these prescalers are non-displaying and are not printed out

EXTERNAL RESET DISABLE - two jumper plugs are available for insertion in desired channel(s) to inhibit response of the module to reset commands received via the Model C-1404 control cable. Thus, the scalers may be reset only by front panel control for cumulative count

PERFORMANCE

CAPACITY - two complete six decade scalers; channel A allows full scale time range of 10^4-1 seconds or minutes, or full scale count range of 10^6-1 counts; channel B allows full scale count range of 10^6-1 counts; the range of each channel is expandable by a factor of 10 via internal selectable decade (÷10) prescalers

COUNT RATE - 25MHz minimum, 30MHz typical

PULSE PAIR RESOLVING TIME - less than 40 nanoseconds for both signal inputs

CRYSTAL TIME BASE ACCURACY - 0.005% over operating temperature range

TEMPERATURE OPERATING RANGE - 0 to 50°C

PRINTOUT SEQUENCE - channel A scaler, channel B scaler

AUTOMATIC RESET - initiated when power is first turned on or after power failed; resets all decade counters to zero

INDICATORS

VISUAL DISPLAY - six solid state numeric indicators for each channel; leading zeros are suppressed

COUNT - front panel light is illuminated during accumulation and extinguished when Stop command is initiated

CONNECTORS

CHANNEL A and B SIGNAL IN - front and rear, BNC, UG-1094/U

CHANNEL A and B GATE IN - front and rear, BNC, UG-1094/U

CHANNEL A and B OVERFLOW - rear panel, BNC, UG-1094/U

GATE OUT - rear panel, BNC, UG-1094/U

CONTROL IN - rear panel, Amphenol, 17-10150

CONTROL OUT - rear panel, Amphenol, 17-20150

POWER

+24V — 0mA +12V — 50mA
-24V — 0mA -12V — 250mA
115VAC — 250mA

PHYSICAL

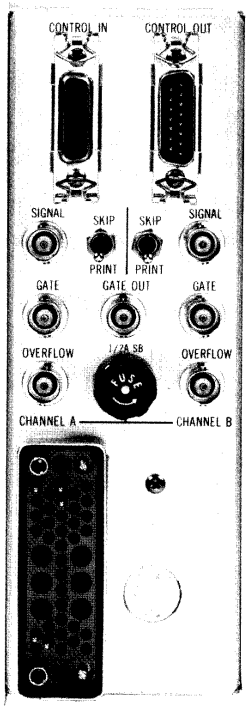
SIZE - standard double width module (2.70 inches wide) (6.85 cm) per TID-20893

NET WEIGHT - 4 lbs (1.82 kgs)

SHIPPING WEIGHT - 9 lbs (4.05 kg)

ACCESSORIES

One Canberra Model C-1404-2 Data/Control Cable 2 ft. in length is supplied for connection to other Canberra data acquisition series modules



DUAL COUNTER/TIMER

Model 1776

Instruction Manual

CANBERRA INDUSTRIES, INC.
45 Gracey Avenue
Meriden, Connecticut 06450

Telephone: (203) 238-2351

9/73

WARRANTY

Canberra-Manufactured Equipment

Equipment manufactured by Canberra Industries, Inc. is warranted against defects in materials and workmanship for a period of twelve months from date of shipment, provided that the equipment has been used in a proper manner as detailed in the instruction manuals. Repairs or replacement, at Canberra's option, will be made without charge at the Canberra factory during this warranty period. Except for the case of defects discovered upon initial operation, shipping expense to Canberra is to be paid by the customer; shipping expense to return the repaired equipment will be paid by Canberra.

The customer must obtain shipping instructions, including an *Authorized Return Number* (ARN), before returning any equipment to the Canberra factory. Compliance with this provision by the customer shall be a condition of this warranty. In giving shipping instructions, Canberra shall not, therefore, assume any liability in connection with the shipment. If, upon receipt of the equipment, Canberra determines that such equipment is not defective within the terms of this warranty, the customer shall pay to Canberra upon invoice, the cost of all transportation and cost of repairs at the then prevailing Canberra repair rate.

This warranty shall not apply to equipment that has been modified or serviced by other than Canberra service personnel, or to failures caused by defective equipment not supplied by Canberra.

This warranty applies only to equipment manufactured by Canberra. On other equipment supplied by Canberra, the full warranty, and only that warranty offered by the original manufacturer, will be passed on to the customer.

WARRANTY ON EQUIPMENT NOT MANUFACTURED BY CANBERRA

Canberra's basic one-year warranty applies only to equipment manufactured by Canberra. Although Canberra may frequently supply, as part of systems, equipment manufactured by other companies, the only warranty that shall apply to such non-Canberra equipment is that warranty offered by the original manufacturer.

Canberra will, upon request, state what warranties are offered by the original manufacturers of such items as computers, teletype machines, printers, plotters, and other non-Canberra equipment which may be supplied as part of a Canberra system. In no case, however, will Canberra assume any liability for such equipment other than to pass on to its customer whatever warranty is supplied by the original manufacturer.

WARRANTY ON SOFTWARE

Canberra will warrant system operation with *Canberra Laboratory Automated Software Systems* (CLASS) only. If the customer decides to use software other than CLASS, Canberra assumes *no* responsibility. Engineering assistance, however, for non-CLASS software is available to the user and should be contracted separately if desired.

ON-SITE WARRANTY OPTION

The basic Canberra warranty applies only to equipment manufactured by Canberra which is *returned to the factory*. If equipment must be repaired at the customer's site, the actual repair labor and parts will be provided at no charge during the warranty period. However, travel expenses to and from the customer's site, and living expenses while on site, shall be paid by the customer unless an on-site warranty option has been purchased. This option may only be purchased prior to shipment of the equipment to the customer.

The on-site warranty option provides for free on-site warranty work (Canberra pays all travel and living expenses) within the first 60 days after delivery of equipment to the customer. If installation is ordered from Canberra, the 60 day period commences upon completion of the initial installation. After the 60 day period, labor and materials used on site will still be covered by the basic warranty, but the customer shall pay for all travel and living expenses incurred for any on-site service.

The price of this 60 day on-site warranty option is $300.00 or two percent (2%) of the entire system list price, whichever is greater.

The on-site warranty option is available only within the contiguous forty-eight (48) United States and Canada.

After the 60 day on-site warranty period, or after initial installation of the equipment, a maintenance contract may be purchased. This is to be contracted through Canberra's Customer Engineering Department. Contact the factory for details concerning maintenance contracts.

INSTALLATION

Installation of equipment purchased from Canberra shall be the sole responsibility of the customer unless it is specifically contracted for at the prevailing Canberra field service rates. To insure timely installation after receipt of equipment, it is recommended that installation be contracted for at the time the equipment is ordered.

REPAIRS

Any Canberra-manufactured instrument no longer in its warranty period may be returned, freight prepaid, to our factory for repair and realignment. When returning instruments for repair, contact the factory for shipping instructions and an *Authorized Return Number* (ARN).

All correspondence concerning repairs should include Model Number and a description of the problem observed.

Once repaired, all equipment passes through our normal pre-shipment checkout procedure, and will meet or surpass its original specifications when returned. Return shipping expense on out-of-warranty repairs will be charged to the customer.

For instruments out of warranty, the customer must supply a purchase order number for the repair before the item will be returned.

SHIPPING DAMAGE

Shipments should be carefully examined when received for evidence of damage caused by shipping. If damage is found, immediately notify Canberra and the carrier making delivery, as the carrier is normally responsible for damage caused in shipment. Carefully preserve all documentation to establish your claim. Canberra will provide all possible assistance in damage claims.

DUAL COUNTER/TIMER

Model 1776

CONTENTS

1 INTRODUCTION 1-1

2 SPECIFICATIONS

2.1 Inputs 2-1
2.2 Outputs 2-1
2.3 Front Panel Controls 2-2
2.4 Rear Panel Controls 2-2
2.5 Internal Controls 2-2
2.6 Performance 2-3
2.7 Indicators 2-3
2.8 Connectors 2-3
2.9 Power 2-3
2.10 Physical 2-3
2.11 Accessories 2-4

3 CONTROLS/INDICATORS

3.1 Front Panel 3-1
3.2 Rear Panel 3-2
3.3 Internal 3-2

4 OPERATING INSTRUCTIONS

4.1 Installation 4-1
4.2 Single Unit Operation 4-1
4.2.1 Manual Control 4-1
4.2.2 Gating 4-2
4.2.3 Preset Count 4-2
4.2.4 Automatic Operation 4-2
4.2.5 Cumulative Operation 4-3
4.3 Non-Printing System Operation 4-3
4.4 Printing System Operation
4.4.1 General 4-3
4.4.2 System Setup 4-4
4.4.3 System Operation 4-4

5 THEORY OF OPERATION

5.1 Input Discriminator 5-1
5.2 Scaler 5-2
5.3 Display, Readout 5-2
5.4 Start/Stop/Gate 5-3
5.5 Timebase 5-3
5.6 Preset Operation 5-3
5.7 Recycle Operation 5-3

DUAL COUNTER/TIMER

Model 1776

Section 1

INTRODUCTION

The Canberra Model 1776 Dual Counter/Timer combines in a dual-width NIM module two independent six decade scalers, plus an internal crystal controlled time base. Two presettable counting channels, identified as A and B, permit preset count and preset time operation either separately or simultaneously. The module may function as a system master to control the duration of counting experiments on the basis of preset count and/or time, or may serve simply as a dual channel system slave, operating at counting rates up to 25MHz.

Independent thumbwheel switches select the preset count or time limit for each channel. The N switch selects a Multiplier (0-9) and the 10^M switch selects a power of ten (10^0-10^5) by which it is multiplied. Other features include: front panel independent electronic gating; integral discrimination (+0.5 to +10 volt range) of both channel A and B input signals; manual START, STOP, and RESET controls; SINGLE and RECYCLE modes of operation; POSITIVE and NEGATIVE input pulse selection on each channel; and a rear panel GATE OUT connector for gating up to 50 compatible scalers and timers.

Accumulated data printout of the two internal scalers is accomplished with standard Canberra Print/Control NIM modules such as the Model 1488 Teletype Scanner. Printout of either channel may be inhibited by rear panel switch. When employed in a printing system, the printout controller generally establishes SINGLE or RECYCLE mode of operation. In a non-printing system, the Model 1776 may be operated in either the SINGLE or RECYCLE mode. In the RECYCLE mode the display time between recycles is internally adjustable from 1 to 5 seconds.

The Model 1776 display utilizes light-emitting diodes to provide solid state reliability and long life. Because this is a flat-surface display, the viewing angle is far wider than that obtained with tube type displays. A circularly polarized filter allows brilliant readout even in direct sunlight. Two sets of six numeric indicators display the contents of Channel A and B independently.

Channel A has the capability of presetting and counting either the internal time base (0.01 sec or 0.01 min.) or the external A count input signals (CNT), depending upon the setting of the front panel Channel A switch labeled .01 min/CNT/.01 sec. Channel B is a second scaler which is presettable and independent of Channel A. The B scaler counts and displays the B count input signals.

Section 2

SPECIFICATIONS

2.1 INPUTS

SIGNAL POS.

Channel A and B; accept +0.5 to +10 volt (+50 volts maximum) pulses
Rise time: any
Pulse width: 20 nanoseconds
Input impedance: greater than 1000Ω; DC coupled; front and rear panel BNC connectors

NEG.

Channel A and B; accept -0.6 to -2.0 volt pulses
Pulse width: 20 nanoseconds minimum
Input impedance: 50Ω; DC coupled; front and rear panel connectors

GATE

Channel A and B; +3 to +10 volt DC level or pulses (or open circuit) enables accumulation; less than 0.8 volt DC level or pulse (or short circuit) inhibits accumulation
Input impedance: greater than 1000Ω; DC coupled, front and rear panel connectors

CONTROL IN

Accepts START, STOP, RESET, PRINT (5 volt negative-going pulses from +5 volt level
Rise time: 500 nanoseconds maximum
Width: 1.0 microsecond minimum
Also accepts HOLD commands (DC level change from +5 volts to +0.5 volts maximum during printout; rear panel Amphenol 17-10150 15-pin connector

2.2 OUTPUTS

OVERFLOW

+5 volt pulses at overflow of sixth decade
Rise time: less than 200 nanoseconds
Width: 5 microseconds DC coupled; rear panel BNC connector

GATE OUT

Provides +5 volt DC level during accumulation, +0.5 volt DC level when not accumulating (except when inhibited by Gate Input)
Rise time: less than 200 nanoseconds; will gate up to 50 compatible scalers and timers
Current sinking: 150mA, DC coupled; rear panel BNC connector

CONTROL OUT

Provides START, STOP, and RESET commands (5 volt negative-going pulses from +5 volt level, rise time 500 nanoseconds maximum, width 1.0 microsecond minimum); data output information in the form of decimal digit serial 1-2-4-8 BCD; logic 1 = +4 volts DC, logic 0 = 0 volts DC; connector also provides "next unit" PRINT command; eighth and successive PRINT command pulses are routed out via logic gate; all control lines except PRINT command are wired directly between Control Input-Control Output connectors; rear panel Amphenol 17-20150 15-pin connector

2.3 FRONT PANEL CONTROLS

START	Manual pushbutton to renew or begin accumulation
STOP	Manual pushbutton to terminate accumulation
RESET	Manual pushbutton to reset all decades to zero; accumulation will begin unless the STOP pushbutton is first depressed
CHANNEL A TIME BASE or COUNT	Three position toggle switch to select 0.01 minute or 0.01 second time base; CNT position permits Channel A to function as a presettable counter
MODE	Toggle switch to select SINGLE or RECYCLE mode of operation upon reaching a preset condition
PRESET	Two digit thumbwheel switches for each channel; selectable preset limit is $N \times 10^M$, N is the 0-9 digit multiplier and 10^M is the power of ten being multiplied; either channel N = 0 is preset OFF, disabling the preset functions of that channel
NEG/POS	Two front panel toggle switches to select input pulse polarity for Channels A and B
DISC	Two front panel screwdriver adjustments to inhibit counting of input pulses to Channels A and B below selected level from +0.5 to 10 volts

2.4 REAR PANEL CONTROLS

PRINT	Two rear panel toggle switches (Channels A and B) to enable (PRINT) or inhibit (SKIP) printout of either or both of the two scalers

2.5 INTERNAL CONTROLS

PRESCALER	Jumper plugs to insert a decade prescaler in the desired channel; these prescalers are non-displaying and are not printed out
EXTERNAL RESET DISABLE	Two jumper plugs may be removed from desired channels to inhibit response of the module to reset commands received via the Model 1404 control cable. Thus, the scalers may be reset only by front panel control for cumulative count

2.6 PERFORMANCE

CAPACITY	Two complete six decade scalers; Channel A allows full scale time range of 10^4-1 seconds or minutes, or full scale count range of 10^6-1 counts; Channel B allows full scale count range of 10^6-1 counts; the range of each channel is expandable by a factor of 10 via internal selectable decade (÷ 10) prescalers
COUNT RATE	25MHz minimum, 30MHz typical
PULSE PAIR RESOLVING TIME	Less than 40 nanoseconds for both signal inputs
CRYSTAL TIME BASE ACCURACY	0.005% stability over operating temperature range; 0.002% setting accuracy
TEMPERATURE OPERATING RANGE	0^{o} to 50^{o}C
PRINTOUT SEQUENCE	Channel A scaler, Channel B scaler
AUTOMATIC RESET	Initiated when power is first turned on or after power failed; resets all decade counters to zero

2.7 INDICATORS

VISUAL DISPLAY	Six solid state numeric indicators for each channel; leading zeros are suppressed
COUNT	Front panel light is illuminated during accumulation and extinguished when STOP command is initiated

2.8 CONNECTORS

CHANNEL A and B SIGNAL IN	Front and rear, BNC, UG-1094/U
CHANNEL A and B GATE IN	Front and rear, BNC, UG-1094/U
CHANNEL A and B OVERFLOW	Rear panel, BNC, UG-1094/U
GATE OUT	Rear panel, BNC, UG-1094/U
CONTROL IN	Rear panel, Amphenol, 17-10150
CONTROL OUT	Rear panel, Amphenol, 17-20150

2.9 POWER

+24V	–	0mA
+12V	–	50mA
- 24V	–	0mA
- 12V	–	250mA
115VAC	–	250mA

2.10 PHYSICAL

SIZE	Standard double width module (2.70 inches wide) per TID-20893 (rev.)

WEIGHT	4 lbs (1.82 kgs)

2.11 ACCESSORIES

One Canberra Model 1404 Data/Control Cable 3 ft. in length is supplied for connection to other Canberra data acquisition series modules

Section 3

CONTROLS/INDICATORS

3.1 FRONT PANEL

3.1.1 START Pushbutton

When depressed, initiates data accumulation in the Model 1776. Scaler is *not* reset to zero.

3.1.2 STOP Pushbutton

When depressed, terminates data accumulation in the Model 1776.

3.1.3 RESET Pushbutton

When depressed, resets scaler contents to zero. The RESET pushbutton does *not* terminate data accumulation.

3.1.4 COUNT Lamp

Illuminates while data accumulation is enabled (between START and STOP commands). The COUNT lamp is *not* inhibited by the GATE input.

3.1.5 SINGLE/RECYCLE Switch

When in RECYCLE position, enables repetitive count cycles, separated by an adjustable (1-5 second) delay. The RECYCLE operation is inhibited if a Print Control module is active in the system.

3.1.6 .01SEC/CNT/.01MIN Switch

Selects input for Channel A scaler.

3.1.7 PRESET Switch (1 per Channel)

Selects preset count (or time), in the form $N \times 10^M$. The first channel to reach preset terminates the counting operation. A preset value of N = 0 in a given channel inhibits preset operation for that channel.

3.1.8 SIGNAL Input (1 per Channel)

Accepts positive or negative pulses for accumulation.

3.1.9 POS/NEG Switch (1 per Channel)

Selects polarity of input pulse to be discriminated. If in NEG position, the SIGNAL input is terminated in 51Ω and the discriminator level is fixed at -0.6 volts. If in POS position, input impedance is 1000Ω, and a variable discriminator level is enabled.

3.1.10 DISC Potentiometer (1 per Channel)

When enabled by POS/NEG switch, selects discriminator level for positive input pulses. Ten-turn pot gives variation from 0.5 to +10 volts.

3.1.11 GATE Input (1 per Channel)

When open (or at a HIGH logic level) enables accumulation of input pulses; when grounded (or receiving a LOW logic level) inhbitis accumulation of input pulses.

3.1.12 DATA DISPLAY

Provides visual readout of six decades of each scaler. If the prescaling decade is employed, it is *not* displayed. Leading zeros are suppressed.

3.2 REAR PANEL

3.2.1 SIGNAL Inputs

Identical to Front Panel inputs.

3.2.2 GATE Inputs

Identical to Front Panel inputs.

3.2.3 OVERFLOW Output (1 per Channel)

Transmits a 5μsec positive pulse on the overflow of most significant decade.

3.2.4 SKIP/PRINT Switch (1 per Channel)

Inhibits or enables printout of Model 1776 when connected to Print Control module.

3.2.5 GATE Output

Transmits a +5 volt level while accumulation is enabled (the logical AND of the COUNT enable and the GATE input). Transmits a 0 volt current sink when not enabled to accumulate.

3.2.6 FUSE

1/2 amp Slo-Blo inserted in 110 VAC input line.

3.2.7 CONTROL INPUT/CONTROL OUTPUT Connectors

Accept Model 1404 data cables for control and printout of Model 1776 via suitable Master or Print Control modules.

The Model 1776 responds to and transmits START, STOP, RESET Commands (with functions identical to Front Panel pushbuttons). The RESET function may be internally inhibited. The Model 1776 also responds to HOLD and PRINT CLOCK commands, transmitting its data on the BCD DATA lines.

Corresponding pins in the CONTROL IN and CONTROL OUT connectors are wired together in parallel, with the exception of PRINT CLOCK.

3.3 INTERNAL

Three options are programmable by jumper within the Model 1776: prescaling decade, external reset inhibit, and RECYCLE delay.

3.3.1 JUMPERS

When the jumpers shown in Figure 3.1 are in the "A7" ("B7") position, a prescaling decade is inserted in the accumulation path of channel A (B). When in the "A6" ("B6") position, the prescaler is bypassed.

When the jumpers shown in Figure 3.1 are in the "ER" position, a RESET command received via the CONTROL IN/CONTROL OUT connector will reset the respective scaler. If the jumpers are in other positions, an external reset is inhibited.

3.3.2 RECYCLE TIME ADJUSTMENT

The potentiometer shown in Figure 3.1 adjusts the time between counting operations in the RECYCLE mode from 1 to 5 seconds.

Section 4

OPERATING INSTRUCTIONS

4.1 INSTALLATION

The Canberra Model 1400 Bin and Power Supply, or other bin and power supply systems conforming with the mechanical and electrical standards set by AEC Report TID-20893 (rev.), will accommodate the Model 1776. The right side cover of the double-width NIM module acts as a guide for insertion of the instrument. Secure the module in place by turning the two front panel captive screws clockwise until finger tight. It is recommended that the NIM bin power switch be OFF whenever the module is installed or removed.

The Model 1776 can be safely operated where the ambient air temperature is between $0^{o}C$ and $+50^{o}C$ ($+120^{o}F$ maximum). Perforations in the top and bottom sides permit cooling air to circulate through the module. When relay rack mounted, along with other "heat generating" equipment, adequate clearance should be provided to allow for sufficient air flow through both the perforated top and bottom covers of the NIM bin.

4.2 SINGLE UNIT OPERATION

4.2.1 MANUAL CONTROL

1. Set the Model 1776 front panel controls as follows:

   CHANNEL A TIME BASE Switch . . . . . .01SEC.
   OPERATION MODE Switch . . . . . .SINGLE
   CHANNEL A & B PRESET Thumbwheel Switches . . . N = 0 (Preset "off")
   CHANNEL A & B POS/NEG Switches . . . . . .NEG

2. Apply power to the Model 1776 by turning the NIM bin Power switch ON. Observe visual data displays show zero for Channel A & B. This is the automatic reset feature which is initiated when power is first applied or after power has failed.

3. Connect the same source of -1 volt pulses (repetitive or random) to both the Channel A & B SIGNAL INPUT BNC connectors (either front or rear panel).

   Interfacing fast rise time logic pulses to the Counter/Timer requires that the user terminate the receiving end of the coaxial cable with its characteristic impedance to eliminate reflections (pulse distortion). Improper termination (or none) may cause double counts, incorrect data.

4. Depress START pushbutton and observe COUNT indicator light. Observe Channel A accumulation of internally generated 0.01 SEC time increments and Channel B input count accumulation.

5. Depress STOP pushbutton. Observe Channel A & B will cease accumulation and COUNTING indicator extinguishes.

6. Depress RESET pushbutton. All decade counters of both Channel A & B will reset to zero. Counting can be renewed by depressing the START pushbutton.

   Momentarily depress the RESET pushbutton while the unit is accumulating data. Observe all Channel A & B decade counters reset to zero, but the counting operation is not terminated.

4.2.2 GATING

The electronic gating capability permits simple coincidence experiments without the need for additional coincidence modules, and the capability of inhibiting the Counter/Timer by external command.

1. Depress START pushbutton to begin count accumulation again.

2. Ground (short-circuit) Channel A GATE INPUT BNC connector (front or rear panel). Observe counting stops in Channel A only. Remove short-circuit from Channel A GATE INPUT and observe Channel A resumes counting.

3. Repeat step 2 for Channel B GATE INPUT BNC connector and observe identical results.

   Gating Notes: (a) Both Channel A & B GATE INPUTS are independent of each other. (b) COUNTING light remains on even though either or both channels are gated off. (c) The rear panel GATE OUTPUT signal is a function of the GATE A input.

4.2.3 PRESET COUNT

1. Enter a Channel A PRESET time of 2 sec (N = 2 and M = 2) with front panel controls set the same as Section 4.2.1, step 1.

2. Begin accumulation by depressing START pushbutton. Both Channel A & B will cease counting upon Channel A reaching its preset time of 2 sec. Elapsed time can be observed in Channel A and accumulated count total in Channel B.

3. Set Channel A preset to N = 0 and enter a Channel B PRESET count of 1000 (N = 1 and M = 3).

4. Depress START pushbutton to initiate count accumulation again. When Channel B reaches its preset count of 1000, both Channel A & B will stop counting. Accumulated count total (1000 counts) can be observed in Channel B (and elapsed time .01 second time increments in Channel A).

   Preset Notes: (a) When either channel preset limit is reached, both Channel A & B scalers will stop counting. (b) Either channel N = 0 is preset OFF position, disabling the preset functions of that channel.

4.2.4 AUTOMATIC OPERATION

Automatic data acquisition by the Model 1776 can free the researcher from time-consuming manual control of the many iterations that may be required in counting experiments.

1. Front panel controls should be set the same as Section 4.2.1, step 1.

2. Set Channel A PRESET time of 10 seconds (N = 1 and M = 3).

3. Set OPERATION MODE toggle switch to RECYCLE position.

4. Begin acquiring data by manually depressing START pushbutton. Both channels will stop counting when Channel A reaches its preset time of 10 seconds. Observe that the Reset and Start functions will be performed automatically when the Display Time elapses (time between recycles is adjustable from 1 to 5 seconds via the internal DISPLAY TIME trim control).

   The automatic Recycle function will continuously repeat until manually terminated by setting the OPERATION MODE switch to SINGLE.

4.2.5 CUMULATIVE OPERATION

The Counter/Timer can be made to accumulate its total counts from cycle to cycle, giving a running total rather than a new total for each counting cycle by setting the internal jumper plugs, as shown in Figure 3-2.

4.3 NON-PRINTING SYSTEM OPERATION

In a Canberra Data Acquisition System, a "master" unit can control system operation. The Model 1776 Dual Counter/Timer fulfills this master role; its START, STOP, and RESET functions can control (manually or automatically) up to 50 compatible scalers and timers.

All modules in the counting system are interconnected with Model 1404 System Print/Control cables as shown in Figure 4-1. The sequence of connection is not important because the System Start, System Stop, and System Reset control lines are common (presented in parallel) to all the modules.

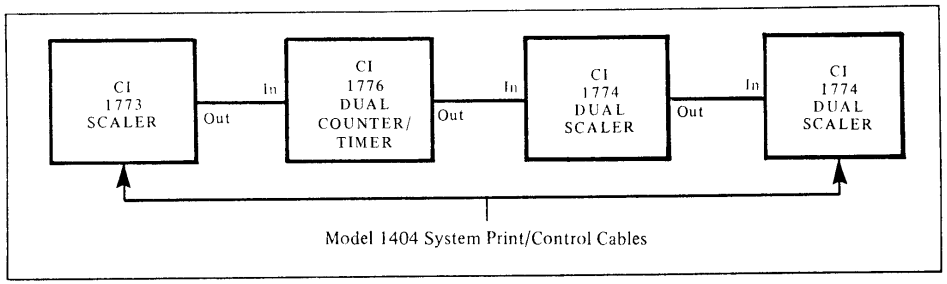


Figure 4-1. Interconnection of Modules for Non-printing System Operation

Operate the Model 1776 as a single unit according to the instructions in Section 4.2. All cable-connected system modules will be simultaneously affected by START, STOP, and RESET commands originating from the Counter/Timer and any other master unit in the counting system.

4.4 PRINTING SYSTEM OPERATION

4.4.1 GENERAL

Four alternatives are available for scanning and recording the data accumulated by a Canberra Data Acquisition System:

Model 1486	ASCII format incremental magnetic tape data scanner.
Model 1487	Computer compatible punched paper tape data scanner.
Model 1488	Punched tape and hard copy Teletype data scanner.
Model 1489	Data scanner/modular strip ("grocery tape") printer.

The task of the Data Scanner is to interrogate each of the scalers and timers in the counting system, receive the incoming data, and prepare it in the proper format for the output recording device.

4.4.2 SYSTEM SETUP

Canberra Data Acquisition Series NIM modules are fully compatible with each other for system control and printout. Up to 50 units can be controlled, scanned, and printed out by a single Data Scanner NIM module.

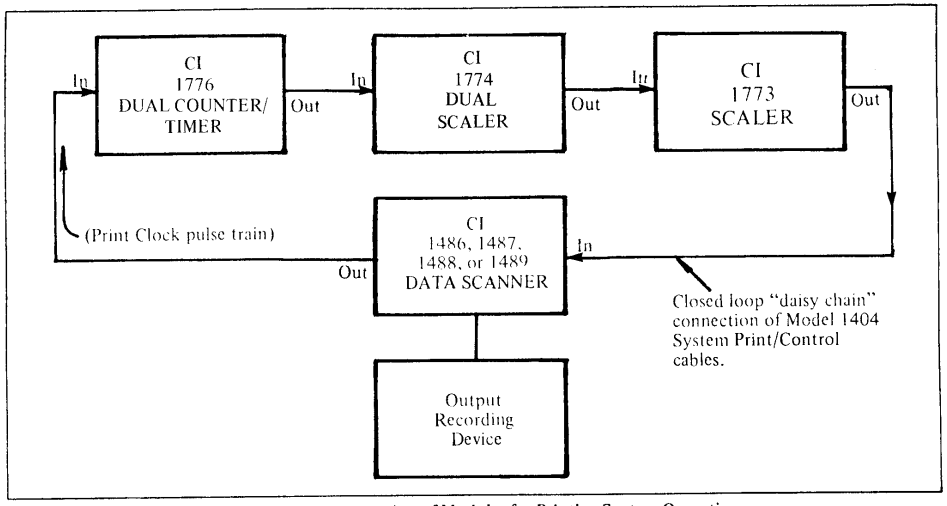


Figure 4-2. Interconnection of Modules for Printing System Operation

All Canberra printing system modules are interconnected by a closed loop "daisy chain" of Model 1404 System Print/Control cables as illustrated in Figure 4-2. Except for the Print Clock line, all control and data lines are directly wired from the CONTROL INPUT to CONTROL OUTPUT connectors on the rear panel of each printing module. A common data/control bus is thus formed, with system commands presented in parallel to all units in the system.

The desired printing sequence is a function of module position in the closed loop system. In Figure 4-2, the Model 1776 will be printed out first, the Model 1774 second, and the Model 1773 last.

Set the Data Scanner's MODE switch to SINGLE and its CONTROL switch to OFF before applying power to the system.

Establish a preset time of 10 seconds on the "master" unit (in this case, the Model 1776) in the system. Master is defined simply as any module in the system that can start, stop or reset the other units in the system.

Set the Model 1776 and Model 1773 rear panel PRINT switches to enable printout of their data contents.

4.4.3 SYSTEM OPERATION

Refer to Figure 4-2 for the following typical printing system operation. In this system both the Model 1776 and Data Scanner function in the master roles, with the Model 1774 and Model 1773 being system slaves.

1. Apply power to the system by turning the NIM bin POWER switch ON.

2. At turn on, all scalers should be reset to 0, and should not be counting.

3. Set the Data Scanner's control switch to ON.

4. Start the system accumulating data by depressing the START pushbutton on the master unit. When the Model 1776 reaches its preset time of 10 seconds, the system will stop counting, and a printout cycle will commence. After printout, the system will await the next manual command.

5. The following sequence of events will help explain what happens during the printout cycle for the counting system in Figure 4-2.

    a. A System STOP command (pin 7) is initiated on the common Data/Control bus by the Model 1776 reaching its preset condition.

    b. The Data Scanner generates a System HOLD command (pin 6) followed by a PRINT CLOCK (interrogate) pulse train (pin 5).

    c. The System HOLD command gates off all visual displays in the printing system (Model 1776 and Model 1773) and removes any previous information from the common BCD Data lines (pins 1, 2, 3, 4).

    d. PRINT CLOCK pulses from the Data Scanner control the sequential steps for printout and continue at a predetermined rate until the printout cycle is completed. The data contents of the module (Model 1776) connected to the CONTROL OUTPUT connector of the Data Scanner is printed out first. Each of the scalers in the Model 1776 is printed out in succession, digit by digit, starting with the most significant data digit in each scaler. As each data digit is printed out, the numeric LED indicator for that digit is illuminated.

    e. After printout, the Model 1776 "enables" (via internal gate logic) succeeding PRINT CLOCK input pulses to be routed out for interrogating the next printing module (Model 1774).

    f. This system printout sequence continues until the last unit (Model 1773) has finished printing. The next PRINT CLOCK pulse passes completely around the closed loop "daisy chain" without finding a unit needing interrogation and returns to the Data Scanner, signalling the end of the printout cycle. All visual displays are then illuminated again.

6. For automatic data acquisition and printout, set the MODE switch on the Data Scanner to RECYCLE. Repeat step 5 of Section 4.4.3. After printout the system will automatically reset, and another accumulate/printout/reset cycle will occur. To stop recycling, set the Data Scanner's MODE switch to SINGLE.

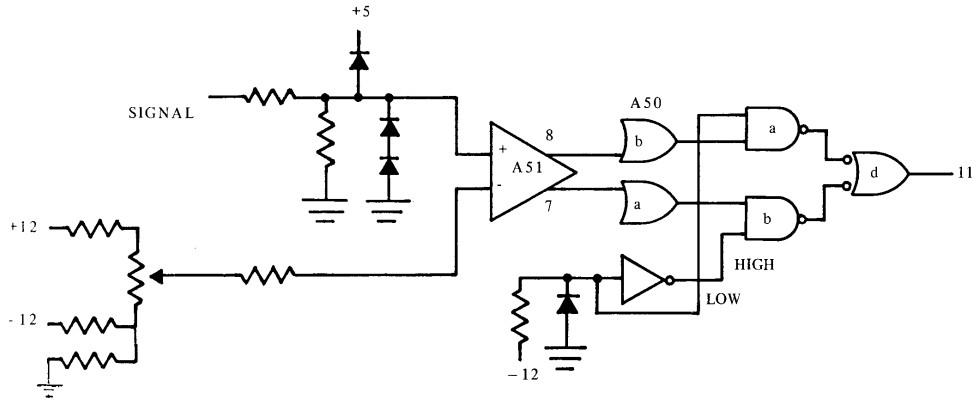
Section 5

THEORY OF OPERATION

5.1 INPUT DISCRIMINATOR

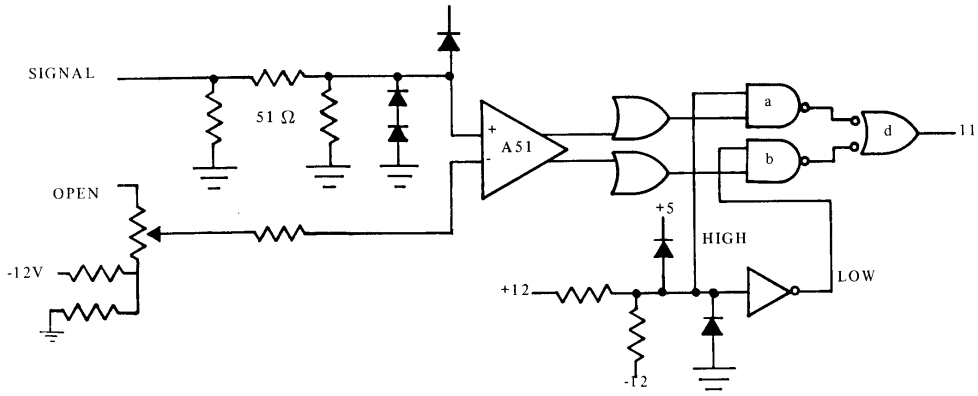
(Identical for Channel A & B; only Channel A is discussed)

Input discriminator A50 is a comparator device which detects negative or positive input pulses as a function of the POS/NEG front panel switch. When the switch is in the POS position, the circuit is configured:



In this configuration, the input pulses are discriminated against the potentiometer setting, and of the two complementary outputs of A50 the output from pin 5 is selected to pass through A49 and A44b; the output from A44, pin 8, is the pulse train to be counted.

If the switch is in the NEG position, the configuration is:

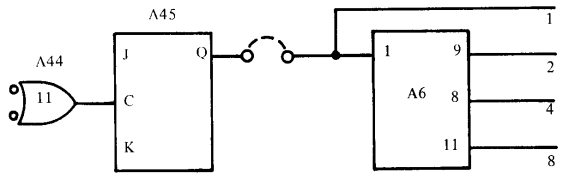


Now the discriminator level is essentially independent of potentiometer setting at approximately -0.5V: and the other output (from A50, pin 7) is selected to become the pulse train to be counted.

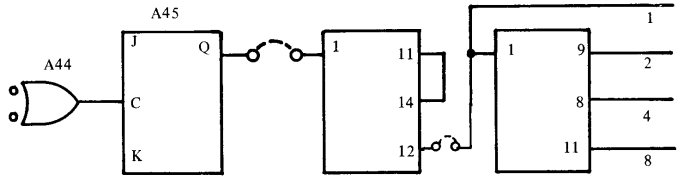
5.2 SCALER

To enable the Model 1776 to count at > 25MHz, the first counting stage is a high speed flip-flop (A45b).

If the prescaler is *not* used, the jumpers are inserted to give the following circuit:



If the prescaler is used, the jumpers give:



The scaler is a standard six stage ensemble, multiplexed for display and readout.

5.3 DISPLAY, READOUT

Under non-readout conditions, oscillator Q18 drives decade counter A75, whose decoded outputs (A69) multiplex scaler data and drive both sets of LEDs. When the "7" count is reached, the counter is reset (via A74, pin 3) to 0.

Leading zero suppression for Channel A is accomplished by A35a. On count 0, the flip-flop is cleared (giving a ripple blanking input). As each digit is strobed, the ripple blanking output is "stored" in A35a. The first non-zero digit sets the flip-flop, forcing the ripple blanking input low to allow display. The flip-flop is always set on the 6th count in any case. A similar procedure is used for Channel B.

When the HOLD line goes low, flip-flop A19b is set (if in the PRINT condition) and the input print clocks control data output via the Model 1404 connector.

The printing operation of the Model 1776 is controlled by the three flip-flops A81a, A81b, and A73a. If either PRINT/SKIP switch is in the SKIP position, the corresponding flip-flop (A81a or b) is held in the reset position. If both are in the SKIP position, the flip-flop A73 is held off, (A72, pin 8 is HIGH) and incoming PRINT CLOCK pulses are gated out.

If, however, at least one of the PRINT/SKIP switches is in the PRINT condition, flip-flop A73a will be set on the leading edge of the HOLD signal (via A74b). Also set will be the flip-flop corresponding to the channel(s) whose PRINT switch is in the PRINT position.

Assume that both switches are in PRINT. The output A81 pin 5 gates the information from Channel A to the data lines: Then as the seventh PRINT CLOCK pulse is received, A81a is reset, and now A80, pin 4, goes HIGH to gate out the information in the B channel. At the seventh PRINT CLOCK pulse, A81b is reset, and A73, pin 6, goes LOW, resetting flip-flop A73a and terminating the readout operation of the Model 1776.

5.4 START/STOP/GATE

Flip-flop A70b/A70c receives the pushbutton outputs, as well as inputs from the Model 1404 connector. This flip-flop enables counting in both channels of the Model 1776, as well as illuminating the COUNT lamp.

Its output (A32, pin 8) is summed with the GATE inputs to enable the J and K inputs to the first counting stage of each channel.

5.5. TIMEBASE

The basic clock for the Model 1776 is derived from a 3.2768MHz crystal oscillator (A68). The 3.2768MHz waveform is passed through two JK flip-flops (inhibited by GATE input and COUNT enable), then through 3 divide-by-16 counters. The resultant waveform is 200Hz, and is divided by 2 (A52) to give 100Hz (0.01 sec. period). The 100Hz signal is divided by 60 to give a waveform of 0.01 min.

If the .01SEC/CNT/.01 MIN switch is in either timebase position, the appropriate waveform is selected (A58), and an edge-triggered clock signal is input to the first counting stage. Additionally, an appropriate decimal point is displayed (via A9, pin 5), and the zero suppressing circuitry is modified to suppress all zeroes to the left of the first integral digit of time; either the 4th or 5th strobe (via A70, pin 13) line is used to reset the suppression flip-flop A35a.

5.6 PRESET OPERATION

The input clock to the scaler, along with the overflow outputs of each stage, are input to an 8-to-1 line multiplexer (A42). The BCD information from the "N" thumbwheel switch on the front panel is used to select one of the multiplexer inputs. The selected input appears at A42, pin 5. This selected waveform is ORed with the START pulse and is used as the "down" clock in A41. When counting is not enabled, counter A41 is loaded with the contents of the "M" thumbwheel switch on the front panel. When counting is enabled, "count" goes HIGH, allowing the "down" clock to decrement the counter A41. When the counter underflows, $\overline{\text{PSTA}}$ goes LOW, firing single-shot A64b to output a system STOP command. If the "M" value is 0, A38, pin 3, will be low, inhibiting the "down" clock from reaching A42 and blocking any preset operation.

5.7 RECYCLE OPERATION

When a STOP command is generated as a result of a preset being reached, A64b momentarily goes LOW, in turn firing the single shot A64a. A64a must be enabled (by the RECYCLE switch in the RECYCLE position). The single shot, A64a, will last for several seconds (as selected by RV1) and will then fire single shot A65b, delivering a RESET/START command to the system.

If a readout cycle results from the Model 1776 reaching preset, the HOLD line will go LOW, and both A64a and A65b will be RESET. Thus the Model 1776 RECYCLE operation will be voided, and the RESET/START command will be generated by the PRINT Module.

Model 2071A
Dual Counter Timer

Features

- 100 MHz count capability for negative inputs and 25 MHz for positive inputs
- Eight-decade count capacity both channels
- Dual counter and timer
- Three-digit preset on time or counts
- Liquid crystal display with annunciators
- Crystal time base
- Adjustable positive input discriminator

Description

The Model 2071A Dual Counter/Timer provides, in a single-width NIM module, two eight-decade counters, a crystal time base, and presetting logic. With more than three operational modes to choose from, complete flexibility has been provided. Normal operation is as a preset timer and event counter, or as a preset event counter and timer. However, the Model 2071A may also operate as a dual counter preset on time without display of the preset.

Both count Inputs accept fast negative NIM or positive pulses, jumper selectable. A discriminator is provided allowing positive inputs to be compatible with unipolar and bipolar linear signals or TTL logic pulses. The front panel positive Discriminator covers an input range of +100 mV to +10 V, permitting selective counting and eliminating spurious counting from noise or other erroneous pulses.

The Model 2071A accommodates input counting rates of 100 million counts per second (100 MHz) for negative inputs and 25 million counts per second (25 MHz) for positive inputs. The negative input pulse pair resolution is 10 ns and will count pulses down to 4 ns in width. The positive input pulse pair resolution is 40 ns and the input pulse width need only be 20 ns above the selected threshold point.

The internal crystal time base provides increments of 0.01 seconds or 0.01 minutes, front panel selectable. Using the appropriate Gate inputs, live time gating with 1 µs resolution can be performed.

A six-digit Liquid Crystal Display (LCD) shows the contents of either counter. The LCD includes eight annunciators which describe the display and dynamic state of the unit. Automatic X100 range shift provides an indication of the six most-significant or six least-significant digits.

Preset control is accomplished with a three-digit thumbwheel switch providing a preset range from one to 99 x 10^6 increments. The preset setting takes the form of NM x 10^P.

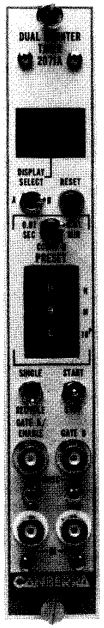
The Single/Recycle mode switch provides a means of automatically placing the unit or system back into the counting mode 10 s or 100 µs, jumper selectable, after preset or termination of the count mode is reached.

Master-Slave system operation can be accomplished by the front or rear panel connection of the Enable signal.

For multiple counting systems, the Enable output from a master 2071A will allow slave 2071A or 2072As to start, stop and reset.

For an accumulation of data in channel A, a jumper can be changed to prevent the resetting of channel A.

External connection to Gate A and Gate B inputs permit gating of respective input pulses when in the preset count mode. Time base gating is jumper selectable from Gate A or Gate B inputs and provides 1 µs resolution. Thus when a system busy signal is connected to this



input, the Model 2071A will provide extremely accurate dead time correction.

Specifications

INPUTS

A and B IN – Front panel BNC connectors accept positive linear, TTL, or negative NIM fast logic signals, jumper selectable. Positive operation accepts unipolar, bipolar, or TTL signals; amplitude: 100 mV to +10 V, +12 V maximum. Input amplitude must exceed the adjustable discriminator level for 20 ns to be counted; negative NIM fast logic amplitude: –0.6 V to –1.6 V; minimum width 4 ns below –0.6 V; input rise and fall times: 1 ns to 300 ns; $Z_{in} \approx 1$ kΩ dc for positive signals and ≈ 50 Ω for negative signals.

GATE B – Counting control for channel B or time channel, jumper selectable; with time gate selected, 1 µs time resolution can be obtained; shipped as channel B counting control. Amplitude: +4 to +5 V or open circuit allows counting; 0 to +0.8 V inhibits counting; $Z_{in} \approx 100$ kΩ to +5 V bus. When used as time gating, system busy can be used for accurate dead time correction.

INPUT/OUTPUT

GATE A/ENABLE – Front panel dual function BNC connector; internal jumper selects either GATE A or ENABLE function; factory set for GATE A.

GATE A – With respect to channel A, same as GATE B.

ENABLE – Bidirectional input or output; open collector function, logic "0" OR; sinks 20 mA, sources from 4.7 kΩ to +5 V bus.
Enable Output – Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables counting, +4 V to +5 V disables counting.
Enable Input – 0 to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.
ENB/OVF – Rear panel dual function BNC connector; internal jumper selects ENB (enable) or OVF (overflow) function; factory set for Enable.
OVF (overflow) – OUTPUTS a +5 V pulse coincident with the eighth decade reset; rise and fall times <200 ns; width 0.3 µs nominal; dc coupled; sinks 5 mA, sources 2.5 mA at 2.5 V. Internal jumper selects channel A or B as source; factory set for Channel A as source.
ENB (enable) – Bidirectional as input or output, open collector function, logic "0" OR; sinks 20 mA, sources from 4.7 kΩ to +5 V bus.
Enable Output – Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables counting, +4 V to +5 V disables counting.
Enable Input – 0 to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.

FRONT PANEL CONTROLS

START/STOP – Toggle switch initiates or terminates a counting sequence.
RESET – Pushbutton resets both counters to zero.
SINGLE/RECYCLE – Toggle switch selects single or recycle mode of operation when preset is reached.
0.01 SEC/COUNT B/0.01 MIN – Channel B time base or count selector; three-position toggle switch to select 0.01 seconds, 0.01 minutes, or the external IN B as the B counter and preset function. When time is selected, Channel A counts external events through IN A. When Count B is selected, Channel A counts time in seconds. An additional capability selected by internal jumpers provides a time preset and event counting in channel A and B. In this case time is not available for display or readout.
DISC A/B – Fifteen-turn screwdriver adjust potentiometer independently sets A and B channel discriminator thresholds between 100 mV and 10 V for positive inputs.
A/B DISPLAY SELECT – Toggle switch selects which counter's contents are displayed on the LCD.
PRESET NM x 10^P – Three-digit thumbwheel switch for channel B preset. M is units, N is tens, P is the power of 10, N and M range from 0 to 9; P has a range of 0 through 6; N = M = 0 disables the Preset function.

PERFORMANCE

CAPACITY – Eight decades per channel, allowing 10^8-1 counts in each.
COUNT RATE – 100 MHz max. negative, 25 MHz positive.
PULSE PAIR RESOLUTION – 10 ns negative, 40 ns positive.
CRYSTAL TIME BASE ACCURACY – ±0.0025% over operating temperature range.
PRESET COUNT CHANNEL – Limited to 1 MHz count rate; pulse width minimum 300 ns.
INDICATORS – Six-digit Liquid Crystal Display with auto (X100) shift for displaying six most-significant digits. Eight annunciators on LCD describe the display. These are: OF (OverFlow), X100, CNT, MIN, SEC, A, B, decimal point. Active counting is indicated by blinking CNT, MIN, or SEC annunciator. Leading zeroes are suppressed.

INTERNAL JUMPERS

Inhibit reset of channel A when starting. In this mode channel A accumulates counts and is reset only by the manual pushbutton.
Select overflow output from channel B instead of channel A.
Permit counting in both channels. In this mode the module may still be preset on time, but the time cannot be read out.
Allow a pulse on ENABLE INPUT to start the module.
Change recycle time from 10 s to 100 µs.
Select GATE A or B to gate time channel.
Select GATE A or ENABLE as the function of the front panel connector.
Select ENABLE or OVERFLOW as the function of the rear panel connector.
Select input A and B to accept positive linear and TTL, or negative NIM signals.

POWER

+24 V – 35 mA	+12 V – 120 mA
–24 V – 0 mA	–12 V – 200 mA

PHYSICAL

SIZE – Standard single width NIM module; 3.43 x 22.12 cm (1.35 x 8.71 in.) per DOE/ER-0457T.
NET WEIGHT – 0.94 kg (2.05 lb).
SHIPPING WEIGHT – 3.2 kg (7.06 lb).

ENVIRONMENTAL

OPERATING TEMPERATURE – 0 to 45 °C.
OPERATING HUMIDITY – 0-80% relative, non-condensing.
Tested to the environmental conditions specified by EN 61010, Installation Category I, Pollution Degree 2.

odel 2071A
al Counter Timer

tures

-88 (GPIB) Bus
MHz count capability for negative inputs and 25 MHz for
ve inputs
ade count capacity both channels
counter and timer
preset on time or counts
crystal display with annunciators
l time base
width NIM
able positive input discriminator

cription

erra Model 2071A Dual Counter/Timer provides, in a
dth NIM module, two eight-decade counters, a crystal time
presetting logic. With more than 3 operational modes to
om, complete flexibility has been provided. Normal operation
set timer and event counter, or as a preset event counter and
vever, the Model 2071A may also operate as a dual counter
time without display of the preset.

t INputs accept fast negative NIM or positive pulses, jumper
. A discriminator is provided allowing positive inputs to be
e with unipolar and bipolar linear signals or TTL logic pulses.
panel positive DISCriminator covers an input range of +100
10 V permitting selective counting, eliminating spurious
from noise or other erroneous pulses.

l 2071A accomodates input counting rates of 100 million
r second (100 MHz) for negative inputs and 25 million
r second (25 MHz) for positive inputs. The negative input
resolution is 10 nsec and will count pulses down to 4 nsec in
positive input pulse pair resolution is 40 nsec and the input
th need only be 20 nsec above the selected threshold point.

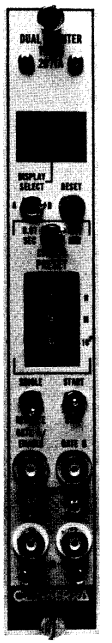
al crystal time-base provides increments of 0.01 seconds or
utes, front panel selectable. Utilizing the appropriate GATE
ve time gating with one microsecond resolution can be
d.

f either counter's contents is accomplished with a six-digit
ystal Display. The LCD includes eight annunciators which
he display and dynamic state of the unit. Automatic X 100
t provides an indication of the six most or six least significant

ntrol is accomplished with a three-digit thumbwheel switch
a preset range from one to 99×10^6 increments. The preset
kes the form of $NM \times 10^P$.

LE/RECYCLE mode switch provides a means of automati-
ing the unit or system back into the counting mode 10
r 100 μsec, jumper selectable, after preset or termination of
mode is reached.

ave system operation can be accomplished via the front or
l interconnection of the ENABLE signal, or by either of the
nal I/O interfaces.



For multiple counting systems, without the 207x-02 NIM Daisy Chain option, the ENABLE output from a master 2071A will allow slave 2071A or 2072As to start, stop and reset.

The Canberra GPIB (IEEE-488) 207x-01 option provides interfacing to a GPIB bus, and allows a GPIB Controller to start, stop, and readout counting systems.

The NIM Daisy Chain 207x-02 option provides interfacing to all existing Canberra printing counters, timers, and readout scanners. It includes a Readout Buffer Capability such that with any number of counting modules a total of only 100 μsec dead time is required. This frees the system to resume counting during readout.

The Canberra EIA Interface 207x-03 option provides for readout to an EIA compatible printer and computer control of start, stop reset and readout via a single cable.

For an accumulation of data in channel A, a jumper can be changed to prevent the resetting of channel A.

al connection to GATE A and GATE B inputs permit gating of ctive input pulses when in the preset count mode. Time base is jumper selectable from GATE A or GATE B inputs and provides sec resolution. Thus when a system busy signal is connected to out, the Model 2071A will provide extremely accurate dead time tion.

ecifications

S

d B IN - Front panel BNC connectors accept positive linear, TTL, egative NIM fast logic signals, jumper selectable. Positive opera- accepts unipolar, bipolar, or TTL signals; amplitude: 0 to +10 V, 2 volts maximum. Input amplitude must exceed the adjustable riminator level for 20 nsec to be counted; negative NIM fast c amplitude: -0.6 V to -1.6 V; minimum width 4 nsec below 6 V; input rise and fall times: 1 nsec to 300 nsec; Z in ≈ 1k ohms or positive signals and ≈ 50 ohms for negative signals.

E B - Counting control for channel B or time channel, jumper ctable; with time gate selected, 1 μsec time resolution can be ained; shipped as channel B counting control. Amplitude: +4 to V or open circuit allows counting; Zero to +0.8 V inhibits nting; Z in ≈ 100 k ohms to +5 V bus. When used as time gating, em busy can be used for accurate dead-time correction.

T/OUTPUT

E A/ENABLE - Front panel BNC connector having a multifunc- capability; GATE A or ENABLE, internally selected with jumper ; shipped in GATE A position.

ATE A - With respect to channel A, same as GATE B.

NABLE - Bidirectional as input or output; open collector func- on, logic "0" OR; sinks 20 mA, sources from 4.7 k ohm to +5 V us.

As an Output - Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables count- ing, +4 V to +5 V disables counting.

As an Input - Zero to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.

B/OVF - Rear panel BNC connector having a multifunction capa- ty; ENABLE or OVERFLOW respectively; internally selected with a per plug; shipped in the ENB position.

VF - OUTPUTS a +5 V pulse coincident with the eighth decade eset; rise and fall times < 200 nsec; width 0.3 μsec nominal; dc oupled; sinks 5 mA, sources 2.5 mA at 2.5 V; internal jumper elects channel A or B as source. Shipped in the channel A osition.

NABLE - Bidirectional as input or output, open collector func- on, logic "0" OR; sinks 20 mA, sources from 4.7 k ohm to +5 V us.

As An Output - Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables count- ing, +4 V to +5 V disables counting.

As an Input - Zero to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.

T PANEL CONTROLS

ART/STOP - Two-position momentary toggle switch initiates or minates a counting sequence.

SET - Manual push-button resets both counters to zero.

IGLE/RECYCLE - Two-position toggle switch selects single or ycle mode of operation upon reaching a preset condition.

1 SEC/COUNT B/0.01 MIN - Channel B time base or count ector; three-position toggle switch to select 0.01 seconds, 0.01 minutes, or the external IN B as the B counter and preset function. When time is selected, Channel A counts external events through IN A. When Count B is selected, Channel A counts time in seconds. An additional capability selected by internal jumpers provides a time preset and event counting in channel A and B. In this case time is not available for display or readout.

A/B DISPLAY SELECT - Two-position toggle switch selects which counter's contents are displayed on the LCD.

PRESET NM × 10^P - Three-digit thumbwheel switch for channel B preset. M is units, N is tens, P is the power of 10, N and M range from 0 to 9; P has a range of 0 through 6; N = M = 0 disables the Preset function.

PERFORMANCE

CAPACITY - Eight decades per channel, allowing 10^8-1 counts in each.

COUNT RATE - 100 MHz max. negative, 25 MHz positive.

PULSE PAIR RESOLUTION - 10 nsec negative, 40 nsec positive.

CRYSTAL TIME BASE ACCURACY - ±0.0025% over operating temperature range.

PRESET COUNT CHANNEL - Limited to 1 MHz count rate; pulse width minimum 300 nsec.

INDICATORS - Six-digit Liquid Crystal Display with auto (x100) shift for displaying six-most significant digits. Eight annunciators on LCD describe the display. These are: OF (Overflow), x100, CNT, MIN, SEC, A, B, decimal point. Active counting is indicated by CNT, MIN, or SEC blinking. Leading zeroes are suppressed.

TEMPERATURE OPERATING RANGE - 0 to 45°C

INTERNAL CONTROLS

STANDARD JUMPERS

- Inhibit reset of channel A when starting. In this mode channel A accumulates counts and is reset only by the manual pushbutton.
- Select overflow output from channel B instead of channel A.
- Permit counting in both channels. In this mode the module may still be preset on time, but the time cannot be read out.
- Allow a pulse on ENABLE INPUT to start the module.
- Change recycle time from 10 sec to 100 μsec.
- Select GATE A or B to gate time channel.
- Select GATE A or ENABLE as function of front panel connector
- Select ENABLE or OVERFLOW as function of rear panel connector.
- Select input A and B to accept positive linear and TTL, or negative NIM signals.

POWER

	with no option	with option 01	with option 02	with option 03
+12 V	120 mA	155 mA	123 mA	190 mA*
-12 V	200 mA*	235 mA*	203 mA*	270 mA*
+24 V	35 mA	35 mA	35 mA	35 mA
-24 V	0 mA	0 mA	0 mA	0 mA

*This current drain exceeds the normal Bin allotment of 167 mA for a single-width module.

PHYSICAL

SIZE - Standard single width NIM module; 3.43 × 22.12 cm (1.35 × 8.71 inches) per TID-20893 (rev.).

NET WEIGHT - 0.94 kg (2.05 lbs.)

SHIPPING WEIGHT - 3.2 kg (7.06 lbs.)

Model 2071A
Dual Counter Timer

Features

- 100 MHz count capability for negative inputs and 25 MHz for positive inputs
- Eight-decade count capacity both channels
- Dual counter and timer
- Three-digit preset on time or counts
- Liquid crystal display with annunciators
- Crystal time base
- Adjustable positive input discriminator

Description

The Model 2071A Dual Counter/Timer provides, in a single-width NIM module, two eight-decade counters, a crystal time base, and presetting logic. With more than three operational modes to choose from, complete flexibility has been provided. Normal operation is as a preset timer and event counter, or as a preset event counter and timer. However, the Model 2071A may also operate as a dual counter preset on time without display of the preset.

Both count Inputs accept fast negative NIM or positive pulses, jumper selectable. A discriminator is provided allowing positive inputs to be compatible with unipolar and bipolar linear signals or TTL logic pulses. The front panel positive Discriminator covers an input range of +100 mV to +10 V, permitting selective counting and eliminating spurious counting from noise or other erroneous pulses.

The Model 2071A accommodates input counting rates of 100 million counts per second (100 MHz) for negative inputs and 25 million counts per second (25 MHz) for positive inputs. The negative input pulse pair resolution is 10 ns and will count pulses down to 4 ns in width. The positive input pulse pair resolution is 40 ns and the input pulse width need only be 20 ns above the selected threshold point.

The internal crystal time base provides increments of 0.01 seconds or 0.01 minutes, front panel selectable. Using the appropriate Gate inputs, live time gating with 1 µs resolution can be performed.

A six-digit Liquid Crystal Display (LCD) shows the contents of either counter. The LCD includes eight annunciators which describe the display and dynamic state of the unit. Automatic X100 range shift provides an indication of the six most-significant or six least-significant digits.

Preset control is accomplished with a three-digit thumbwheel switch providing a preset range from one to 99 x 10^6 increments. The preset setting takes the form of NM x 10^P.

The Single/Recycle mode switch provides a means of automatically placing the unit or system back into the counting mode 10 s or 100 µs, jumper selectable, after preset or termination of the count mode is reached.

Master-Slave system operation can be accomplished by the front or rear panel connection of the Enable signal, or by the optional 207x-03 I/O interface.

For multiple counting systems, the Enable output from a master 2071A will allow slave 2071A or 2072As to start, stop and reset.

The Model 207X-03 EIA Interface option provides for readout to an EIA compatible printer and computer control of start, stop, reset and readout via a single cable.

For an accumulation of data in channel A, a jumper can be changed to prevent the resetting of channel A.

External connection to Gate A and Gate B inputs permit gating of respective input pulses when in the preset count mode. Time base gating is jumper selectable from Gate A or Gate B inputs and provides 1 µs resolution. Thus when a system busy signal is connected to this input, the Model 2071A will provide extremely accurate dead time correction.

Specifications

INPUTS

A and B IN – Front panel BNC connectors accept positive linear, TTL, or negative NIM fast logic signals, jumper selectable. Positive operation accepts unipolar, bipolar, or TTL signals; amplitude: 100 mV to +10 V, +12 V maximum. Input amplitude must exceed the adjustable discriminator level for 20 ns to be counted; negative NIM fast logic amplitude: –0.6 V to –1.6 V; minimum width 4 ns below –0.6 V; input rise and fall times: 1 ns to 300 ns; $Z_{in} \approx 1$ kΩ dc for positive signals and ≈ 50 Ω for negative signals.

GATE B – Counting control for channel B or time channel, jumper selectable; with time gate selected, 1 µs time resolution can be obtained; shipped as channel B counting control. Amplitude: +4

to +5 V or open circuit allows counting; 0 to +0.8 V inhibits counting; $Z_{in} \approx 100$ kΩ to +5 V bus. When used as time gating, system busy can be used for accurate dead time correction.

INPUT/OUTPUT

GATE A/ENABLE – Front panel dual function BNC connector; internal jumper selects either GATE A or ENABLE function; factory set for GATE A.

GATE A – With respect to channel A, same as GATE B.

ENABLE – Bidirectional input or output; open collector function, logic "0" OR; sinks 20 mA, sources from 4.7 kΩ to +5 V bus.

Enable Output – Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables counting, +4 V to +5 V disables counting.

Enable Input – 0 to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.

ENB/OVF – Rear panel dual function BNC connector; internal jumper selects ENB (enable) or OVF (overflow) function; factory set for Enable.

OVF (overflow) – OUTPUTS a +5 V pulse coincident with the eighth decade reset; rise and fall times <200 ns; width 0.3 µs nominal; dc coupled; sinks 5 mA, sources 2.5 mA at 2.5 V. Internal jumper selects channel A or B as source; factory set for Channel A as source.

ENB (enable) – Bidirectional as input or output, open collector function, logic "0" OR; sinks 20 mA, sources from 4.7 kΩ to +5 V bus.

Enable Output – Provides counting control to other 2071A, 2072A or compatible modules; 0 to +0.8 V level enables counting, +4 V to +5 V disables counting.

Enable Input – 0 to +0.8 V enables counting mode; +4 V to +5 V or open circuit disables counting mode.

FRONT PANEL CONTROLS

START/STOP – Toggle switch initiates or terminates a counting sequence.

RESET – Pushbutton resets both counters to zero.

SINGLE/RECYCLE – Toggle switch selects single or recycle mode of operation when preset is reached.

0.01 SEC/COUNT B/0.01 MIN – Channel B time base or count selector; three-position toggle switch to select 0.01 seconds, 0.01 minutes, or the external IN B as the B counter and preset function. When time is selected, Channel A counts external events through IN A. When Count B is selected, Channel A counts time in seconds. An additional capability selected by internal jumpers provides a time preset and event counting in channel A and B. In this case time is not available for display or readout.

DISC A/B – Fifteen-turn screwdriver adjust potentiometer independently sets A and B channel discriminator thresholds between 100 mV and 10 V for positive inputs.

A/B DISPLAY SELECT – Toggle switch selects which counter's contents are displayed on the LCD.

PRESET NM x 10^P – Three-digit thumbwheel switch for channel B preset. M is units, N is tens, P is the power of 10, N and M range from 0 to 9; P has a range of 0 through 6; N = M = 0 disables the Preset function.

PERFORMANCE

CAPACITY – Eight decades per channel, allowing 10^8-1 counts ir each.

COUNT RATE – 100 MHz max. negative, 25 MHz positive.

PULSE PAIR RESOLUTION – 10 ns negative, 40 ns positive.

CRYSTAL TIME BASE ACCURACY – ±0.0025% over operating temperature range.

PRESET COUNT CHANNEL – Limited to 1 MHz count rate; pulse width minimum 300 ns.

INDICATORS – Six-digit Liquid Crystal Display with auto (X100) shift for displaying six most-significant digits. Eight annunciators on LCD describe the display. These are: OF (OverFlow), X100, CNT, MIN, SEC, A, B, decimal point. Active counting is indicated by blinking CNT, MIN, or SEC annunciator. Leading zeroes are suppressed.

TEMPERATURE OPERATING RANGE – 0 to 45 °C.

INTERNAL JUMPERS

Inhibit reset of channel A when starting. In this mode channel A accumulates counts and is reset only by the manual pushbutton.

Select overflow output from channel B instead of channel A.

Permit counting in both channels. In this mode the module may still be preset on time, but the time cannot be read out.

Allow a pulse on ENABLE INPUT to start the module.

Change recycle time from 10 s to 100 µs.

Select GATE A or B to gate time channel.

Select GATE A or ENABLE as the function of the front panel connector.

Select ENABLE or OVERFLOW as the function of the rear panel connector.

Select input A and B to accept positive linear and TTL, or negative NIM signals.

OPTION

Model 207x-03 EIA Interface.

POWER

	with no option	with option 270x-03
+12 V	120 mA	190 mA*
−12 V	200 mA	270 mA*
+24 V	35 mA	35 mA
−24 V	0 mA	0 mA

*This current drain exceeds the normal Bin allotment of 167 mA for a single-width module.

PHYSICAL

SIZE – Standard single width NIM module; 3.43 x 22.12 cm (1.35 x 8.71 in.) per DOE/ER-0457T.

NET WEIGHT – 0.94 kg (2.05 lb).

SHIPPING WEIGHT – 3.2 kg (7.06 lb).

Model 207x-03
EIA Interface

Features

- Direct 8 Digit Readout to EIA Device
- Chain 1-99 Units
- Multiplex Independent Counting Systems
- Complete Control by Terminal or Computer

Description

The Model 207x-03 EIA Interface connects the Models 2071A and 2072A directly to the EIA port of printers or computers, enabling the host unit to communicate with other units and to form counting systems. It is field installable and is compatible with the earlier Models 2071 and 2072.

The 207x-03 is a microprocessor based RS-232C readout device which is compatible with the 2071A and 2072A counters. It is programmed to interface to a wide variety of RS-232C compatible output devices including printers, terminals and computers. The unit automatically assigns identification numbers to each counter in the system and the I.D. is read out along with the data. When connected to an intelligent device such as a microcomputer, the system can be controlled externally and read out while counting. Several independent counting systems (each with its own system master such as a 2071A) can be multiplexed by using a 270x-03 in each counter.

The following modes of operation are possible with the 270x-03:

A. Single Counting System
1-99 counters controlled by a single master.

B. Multiplexed Counting System
2-99 independent counting systems each having up to 99 counters multiplexed into one output device. Systems are formed by connecting the ENABLE signals of all counters in each independent system.

C. Computer Control
External control of Start, Stop, Reset, and Readout of each unit independently by means of computer or terminal.

Counter identification numbers are automatically assigned to units in order of their position in the counting chain when the system is powered up.

The 207x-03 employs a hard-wired chain. Standard chains are available for single and for 1-4 counter systems. Chains for larger and/or multiplexed systems are available on a build-to-order basis.

Specifications

INPUTS
Plug compatible with output signals from 2071A, 2072A series counters and with the earlier 2071 and 2072 counters.

OUTPUT
Standard RS-232C data compatible with EIA devices. XON/XOFF or hardware handshake modes are provided. 25-pin female Dataphone compatible connector.

PERFORMANCE
BAUD RATE – 150 to 19 200 (DIP switch selectable).
WORD LENGTH – 8 digits.
FORMAT – Unit number and two words per counter with selectable termination characters.
PROTOCOL – Data flow control by hardware line or XON/XOFF protocol. Programmable pause after transmitted characters.
CHAIN LENGTH – 1 to 99 units.

CONTROLS
BAUD RATE – Board-mounted DIP switch.
FORMAT – Disable of Carriage Return (CR), Line Feed (LF), Form Feed (FF), or Space.

PHYSICAL
P.C. card with mounting hardware compatible with 2071A, 2072A series counters. Field installable.

CABLE
Not included. Order C1535 for single module or C1536 for up to four modules. Cables for larger systems are available by special order.

POWER
+12 V – 70 mA –12 V – 70 mA

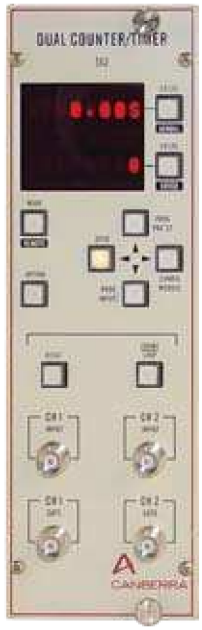
CANBERRA

Model 512
Dual Counter/Timer

Features

- >250 MHz count rate capability, 10^{15}–1 count capacity
- Complete computer control via RS-232 interface
- Standalone programming capability
- High resolution timer with 0.1 µs preset capability
- Self test mode
- Up to eight user-defined operating configurations

Description



The Model 512 Dual Counter/Timer offers greater than 250 MHz count rate capability for the most demanding counting applications. In addition, the 512's counters have a count capacity of 10^{15}–1. Photon counting applications, or experiments requiring pulse-pair resolutions of about 4 ns, are now no longer limited by the counter/timer.

The Model 512 offers standalone performance in addition to being completely computer-controllable. Standalone operations are menu driven, and illuminated pushbuttons are used to select displayed data, operating mode, count/stop/reset operation, and for manual programming. All functions, including threshold settings, are programmable through either the instrument's front panel or the RS-232 interface.

The Model 512 also includes two timers. The non-gateable standard timer has a selectable resolution of 0.01 seconds or 0.01 minutes, and is programmable from 0.01 to 99 999 999.99 seconds/min. The standard timer is software-based, so both hardware counting channels are available for counting.

The high resolution timer is gateable and fully programmable, with a resolution of 0.1 µs from 1 µs to 9.999 999 9 seconds, and 10 s resolution up to 99 999 990.0 seconds. Depending on the mode of operation, the high resolution timer uses channel 1 or channel 2, so that only one counting channel is available when the high resolution timer is used. Also included is a programmable recycle timer which resets and initiates a new counting interval at the completion of a cycle.

The Model 512 includes two high speed counters capable of greater than 250 MHz operation. Polarity, threshold, and counter/timer presets are all fully front-panel programmable. Depending on the mode of operation, channel 1 may be defined as a preset counter or preset high resolution timer. Also, depending on the mode selected, channel 2 may be defined as a high resolution interval timer. When channel 1 is defined as the preset counter, it is fully programmable from 1 to 99 999 999 counts. Both counter channels have a dynamic range of 10^{15}–1 counts, so overflow outputs are not required. Both channels are gateable from the front-panel GATE inputs in both counter and timer configurations.

The Model 512 includes a software-based event counter which increments at the completion of each counting interval. If the event counter value equals the event preset register, further count intervals are inhibited (recycle is disabled).

Phone contact information

Benelux/Denmark (32) 2 481 85 30 • **Canada** 905-660-5373 • **Central Europe** +43 (0)2230 37000 • **France** (33) 1 39 48 52 00 • **Germany** (49) 6142 73820
Japan 81-3-5844-2681 • **Russia** (7-495) 429-6577 • **United Kingdom** (44) 1235 838333 • **United States** (1) 203-238-2351

For other international representative offices, visit our web site: **http://www.canberra.com** or contact the CANBERRA U.S.A. office.

C9669 3/07 Printed in U.S.A.

Model 512 Dual Counter/Timer

Eight different user-defined operating configurations: presets, input polarity/threshold, operating modes, etc., may be saved in non-volatile memory and recalled on command. In addition, one of these configurations may be selected as the power-on default configuration. A self-test mode allows confirmation of proper counter/timer functioning at power on.

The built-in RS-232 serial communications interface provides computer control of all Model 512 functions except the communications parameters.

Specifications

INPUTS

- INPUTS (CH1/CH2) – Polarity and threshold are selected through the PROG INPUTS function; input must exceed threshold; when polarity is changed, the threshold defaults to the appropriate NIM logic level threshold: +1.5 V (pos) or –250 mV (neg); front panel BNC connectors.
  - Negative polarity: Z_{in}=50 Ω; dc coupled; amplitude: –100 mV to –5 V; minimum pulse width >2 ns; threshold is adjustable from –100 mV to –5.0 V; resolution = 5 mV.
  - Positive polarity: Z_{in}=3 kΩ; dc coupled; amplitude: +100 mVto +10 V, minimum pulse width >20 ns; threshold is adjustable from +100 mV to +10.0 V; resolution = 5 mV.
- GATES (CH1/CH2) – Accept NIM positive logic signals to enable/disable counting in the channel. Open circuit or V_{in} >+2.4 V enables counting.V_{in} <+1.5 V disables counting; front panel BNC connectors.

INTERVAL IN/OUT

- This connector controls or is controlled by the master count interval. In the counters only mode, this connector is an input; in all other modes, it is an output; rear panel BNC connector.
  - IN – Receives an external interval signal: open circuit, or V_{in} >+2.4 V, allows counting; V_{in} <+1.5 V stops counting; TTL-compatible input.
  - OUT – Provides a nominal +5 V during the counting interval and <+1.5 V when counting is disabled. I_{sink} and I_{source} ≈30 mA, maximum, for NIM positive logic levels.

RS-232

- Standard RS-232C serial communications connector; rear panel DB-25 connector.

DISPLAY

- DISPLAY – Data display features automatic formatting and leading zero suppression. The display shifts to an exponential format when accumulated counts exceed 10^8–1 counts, or displayed time is greater than 10^4–1 s/min. In addition, using the high resolution timer and displaying intervals of less than one second results in a ms or µs display. The full display can be restored by using the LEFT/RIGHT cursor buttons. Two front panel 8-digit alphanumeric dot-matrix displays.

CONTROLS

- CH SEL/SCROLL – In local or remote, CH SEL is used to select one channel for display from all available for that mode. In local, SCROLL is used to scroll through menu selections during programming. Scroll is disabled in remote. Front panel pushbutton switch with LED.
- CHSEL/ENTER – In local or remote,CHSEL is used to select one channel for display from all available for that mode. In local, ENTER is used to select an item from the menu or to enter new presets, thresholds, etc. Enter is disabled in remote. Front panel pushbutton switch with LED.
- MODE/REMOTE – In local, pressing MODE enables the OPER, PROG PRESET, CONFIG MODULE, and PROG INPUTS pushbuttons for two seconds. In remote, the pushbutton is illuminated continuously to indicate computer control. Front panel pushbutton switch with LED.
- LEFT/RIGHT – In local and remote, pressing the LEFT and RIGHT buttons deformats the display for 20 s and scans the display left and right. In local only, enters presets and thresholds during programming. Front panel pushbutton switch with LED.
- UP/DOWN– In local, used to enter presets and thresholds during programming. Front panel pushbutton switch with LED.
- OPER – In local, MODE+OPER returns the unit to normal operation from a programming mode; enables auto-formatting of displays; front panel pushbutton switch with LED.
- PROG PRESET – In local, MODE+PROG PRESET enters the Presets Programming mode; front panel pushbutton switch with LED.
- PROG INPUTS – In local, MODE+PROG INPUTS enters the Inputs Programming Mode; front panel pushbutton switch with LED.
- CONFIG MODULE – In local, MODE+CONFIG MODULE enters the Configure Module Programming mode; front panel pushbutton switch with LED.
- OPTION – In local, selects available functions; front panel pushbutton switch with LED.
- RESET – In local, RESET+COUNT/STOP resets counters and timers and stops counting; front panel pushbutton switch with LED.
- COUNT/STOP – In local, starts or stops counting; synchronized to timer/counter 1, depending on the mode; front panel pushbutton switch with LED.

PERFORMANCE

- COUNT CAPACITY – Counter channels have a count capacity of 10^{15}–1 counts. The Preset Counter channel has a capacity of 10^8–1 counts.
- MAXIMUM COUNTING RATE – Maximum counting rate for positive inputs is 25 MHz. For negative inputs, maximum rate is greater than 250 MHz.
- PULSE PAIR RESOLUTION – <40 ns for positive inputs; <4 ns for negative inputs.

Model 512 Dual Counter/Timer

- TIME BASE – Two time bases are provided, depending on mode.
  - Standard timer: resolution is selectable for 0.01 s or min; programmable from 0.01 s or min to 99 999 999.99 s or min.
  - High resolution timer: resolution of 0.1 µs from 1 µs to 9.999 999 9 s; resolution of 10 s up to 99 999 990.0 s.
- TIME BASE INACCURACY – Maximum error for both time bases is ≤±0.0025% inclusive of calibration tolerance at 25 °C, operating temperature range, power supply variation, aging, shock and vibration.

POWER REQUIREMENTS

+12 V – 110 mA	+6 V – 410 mA
–12 V – 10 mA	–6 V – 800 mA

PHYSICAL

- Standard double-width NIM module; 6.86 x 22.12 cm (2.70 x 8.71 in.) per DOE/ER-00457T.
- NET WEIGHT – 1.36 kg (3.0 lb).
- SHIPPING WEIGHT – 2.27 kg (5.0 lb).

ENVIRONMENTAL

- OPERATING TEMPERATURE RANGE – 0–50 °C.
- OPERATING HUMIDITY – Up to 95%, non-condensing.

CE

ISO 9001 SYSTEM CERTIFIED

©2007 Canberra Industries, Inc. All rights reserved.

Model 1788
Communications Interface

Features

- Fully EIA RS-232C compatible
- Variable speed selection from 300 Baud to 9600 Baud

Description

The Model 1788 Communications Interface combines a Canberra "Daisy Chain" readout scanner and EIA RS-232C interface in a double width NIM module, to enable a computer to control the Canberra Daisy Chain.

Internal jumpers are used to select the data transmission rates. An internal crystal oscillator is used to provide a range of 300-9600 Baud. In addition, an external oscillator may be used to provide any desired rate over the range of 0-20,000 Baud. The standard selectable rates are:

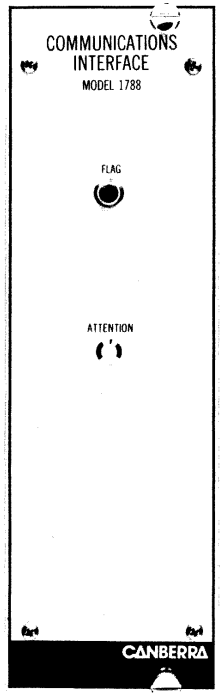
Baud Rate	Time to Read one Scaler
300	265ms
1200	66ms
2400	33ms
4800	16ms
9600	8.5ms

Other internal jumpers are provided to allow the user to select the existence and type of parity to be generated, and to select computer or modem operation.

All communication between the Model 1788 and the computer is via printing ASCII characters. This includes system control as well as data transfer. The ASCII control characters and their meaning are as follows:

Commands from 1788 to Computer		DESCRIPTION
!	$(041)_8$	The counting system has given a STOP (reached Preset) or, if a data transfer is in progress, the last scaler has all ready been readout
"	$(042)_8$	Generated by the front panel pushbutton
#	$(043)_8$	Generated by rear panel BNC input pulse or logic level

Commands from Computer To 1788		
R	$(122)_8$	Issue a system RESET
A	$(101)_8$	Illuminate the 1788 front panel lamp
S	$(123)_8$	Issue a system START
B	$(102)_8$	Turn off the 1788 front panel lamp
H	$(110)_8$	Issue a system STOP
X	$(130)_8$	Issue a logic pulse from the 1788 rear panel BNC
I	$(111)_8$	Force the system into PRINT and/or readout one scaler



The following is an example of how the above control characters are used

Computer → 1788	1788 → Computer	Action
R		RESET System
S		START system, then wait
	!	Preset has been reached
I	Scaler #1 Data	PRINT initiated and first data transferred
I	Scaler #2 Data	Data from second scaler
I	Scaler #3 Data	Data from third scaler
I	!	Transfer complete; there were only 3 scalers
X		Advance sample changer and wait
	#	Sample changer ready
R		RESET system
etc.		

From the above example, the ease with which the Model 1788 can be controlled becomes apparent, allowing the experimenter to easily upgrade an off-line data acquisition system into an on-line acquisition and analysis system.

Specifications

INPUTS

CONTROL IN - Canberra "Daisy Chain" input; rear panel

SIGNAL (#) - Rear panel BNC; a low-to-high or high-to-low level shift (jumper selectable) will transmit a "#" to the computer; input impedance ≈3K ohms; a low voltage is - 15VDC to 0.8VDC; a high voltage is +3VDC to 15VDC

EXTERNAL RATE - Rear panel BNC; a TTL input for externally determining the Model 1788 Baud rate; input must be 16 times the desired rate

OUTPUTS

CONTROL OUT - Rear panel; Canberra "Daisy Chain" output

SIGNAL (X) - Rear panel BNC, Computer generated output; internal jumpers allow for

a. 4μsec Low to High pulse
b. 4μsec High to Low pulse
c. Level state (Low or High) corresponding to the state of the front panel indicator lamp

EIA DATA - Rear panel EIA RS-232C connector

CONTROLS/INDICATORS

FLAG - Front panel pushbutton which transmits a "character" to the computer

ATTENTION - Front panel indicator controlled by the computer

CONNECTORS

CONTROL INPUT - Rear panel, Amphenol # 17-10150
SIGNAL (#) - Rear panel BNC, # UG-1094/U
EXTERNAL RATE - Rear panel BNC, # UG-1094/U
CONTROL OUTPUT - Rear panel, Amphenol # 17-20150
SIGNAL (X) - Rear panel BNC, # UG-1094/U
DATA - Rear panel, standard EIA RS-232C connector

POWER

+24V	–	0mA	+12V	–	35mA
- 24V	–	0mA	- 12V	–	20mA

115VAC – 70mA (rear panel 1/2 amp S.B. fuse and holder)

PHYSICAL

SIZE - Standard double-width NIM module (2.70 × 8.71 inches) (6.85 × 22.19 cm) TID 20893 (rev.)
NET WEIGHT - 4.1 lbs. (1.86 kg)
SHIPPING WEIGHT - 9.1 lbs. (4.10 kg)

ACCESSORIES

One Canberra Model C1404 Data/Control cable 3 ft. in length is supplied with unit.

Φ15Φ

CANBERRA

ASYNCHRONOUS LINE INTERFACE
for
C.I. DATA SYSTEM MODULES

Model 1788
Instruction Manual
December 1978

ASYNCHRONOUS LINE INTERFACE
for
C.I. DATA SYSTEM MODULES

Model 1788
Instruction Manual
December 1978

CANBERRA INDUSTRIES, INC.
45 Gracey Avenue
Meriden, Connecticut 06450

Telephone: (203) 238-2351

Educational Services

In addition to Canberra's installation and service support, Canberra Industries, Inc. Educational Services provides a complete Training Program. This program provides both Operations/Software and Hardware Maintenance Training for new system users. For further information on any of our courses contact:

Marketing Services
Canberra Industries
45 Gracey Avenue
Meriden, Ct. 06450

Phone: (203)238-2351 Extension: 208

M41R1 12/78 Printed in U.S.A. ©Copyright 1978 By Canberra Industries, Inc., Meriden, Ct.

BASIC WARRANTY

CANBERRA - MANUFACTURED EQUIPMENT

Equipment manufactured by Canberra Industries, Inc. is warranteed against defects in materials and workmanship for a period of twelve months from date of shipment, provided that the equipment has been used in a proper manner as detailed in the instruction manuals. During the warranty period, repairs or replacement will be made at Canberra's option on a return to factory basis. The transportation cost, including insurance, to and from Canberra, is the responsibility of the Customer. Except for defects discovered upon initial operation, shipping expense to Canberra is to be paid by the customer; shipping expense to return the repaired equipment will be paid by Canberra.

The customer must obtain shipping instructions, including an Authorized Return Number (ARN), before returning any equipment to the Canberra factory. *Compliance with this provision by the customer shall be a condition of this warranty.* In giving shipping instructions, Canberra shall not be deemed to have assumed any responsibility or liability in connection with the shipment. If, upon receipt of the equipment, Canberra determines that such equipment is not defective within the terms of this warranty, the customer shall pay to Canberra, upon invoice, the cost of diagnosis at the then prevailing Canberra repair rate and the cost of return transportation.

The Canberra Basic Warranty applies only to equipment manufactured by Canberra which is returned to the factory. If equipment must be repaired at the customer's site, the actual repair labor and parts will be provided at no charge during the warranty period. However, travel expenses to and from the customer's site, and living expenses while on site, shall be paid by the customer unless an On-Site Warranty Option has been purchased. This option may only be purchased prior to shipment of the equipment to the customer.

This warranty shall not apply to Canberra equipment that has been modified or serviced by other than Canberra Service Personnel, or to failures of Canberra equipment caused by defective equipment not manufactured by Canberra.

The Express warranties set forth herein are the only warranties with respect to the products, or any materials or components purchased from others and furnished by Canberra, and there are no other warranties, expressed or implied. The warranty of merchantibility is expressly limited as herein provided and all warranties of fitness are expressly disclaimed and excluded. Canberra shall have no liability for any special, indirect or consequential damages, whether from loss of production or otherwise, arising from any breach of warranty hereunder or defect or failure of any product or products sold hereunder.

EXCLUSIONS

Warranty service is contingent upon the proper use of all equipment and does not cover equipment which has been modified without Canberra's written approval or which has been subjected to unusual physical or electrical stress as determined by Canberra Service personnel. Canberra Industries shall be under no obligation to furnish warranty service (preventive or remedial): (1) if adjustment, repair or parts replacement is required because of accident, neglect, misuse, failure of electrical power, air conditioning, humidity control, transportation, or causes other than ordinary use: (2) if the equipment is maintained or repaired or if attempts to repair or service equipment are made by other than Canberra personnel without the prior approval of Canberra.

This warranty does not cover detector damage caused by warm-up or by neutrons or heavy charged particles. Damage from these causes is readily identifiable as described in the manual accompanying each detector.

EQUIPMENT NOT MANUFACTURED BY CANBERRA

Canberra's basic one-year warranty applies only to equipment manufactured by Canberra. Although Canberra may frequently supply, as part of systems, equipment manufactured by other companies, the only warranty that shall apply to such non-Canberra equipment is that warranty offered by the original manufacturer if any.

Canberra will, upon request, offer, as an option, warranty coverage for non-Canberra equipment such as computers and peripherals sold as part of a system supplied by Canberra. Quotations on this coverage may be obtained by contacting Canberra Nuclear Systems Division.

SOFTWARE

Canberra warrants proper system operation *only* with programs developed by Canberra using the operating system supplied to the customer. Canberra assumes no responsibility for user-written programs or programs published as part of information exchange in Canberra periodicals.

Engineering assistance for software development is available and can be contracted through the Canberra Nuclear Systems Division Sales Department.

INSTALLATION

Installation of equipment purchased from Canberra shall be the sole responsibility of the customer unless the installation is specifically contracted for at the prevailing Canberra field service rates. To insure timely installation after receipt of equipment, it is recommended that installation be contracted for at the time the equipment is ordered.

ON-SITE WARRANTY OPTION

The On-Site Warranty Option provides for free on-site warranty work (Canberra pays all travel and living expenses) within the first 90 days after delivery of equipment to the customer. If installation is ordered from Canberra, the 90 day period commences upon completion of the initial installation. After the 90 day period, labor and materials used on site will still be covered by the basic warranty, but the customer shall pay for all travel and living expenses incurred for any on-site service.

A maintenance contract may be purchased covering the period after the 90 days on-site warranty period, or after initial installation of the equipment. This is to be contracted through Canberra's Nuclear Systems Division.

REPAIRS

Any Canberra-manufactured instrument no longer in its warranty period may be returned, freight prepaid, to our factory for repair and realignment. When returning instruments for repair, contact the Customer Service Department for shipping instructions and an Authorized Return Number (ARN).

All correspondence concerning repairs should include Model Number and a description of the problem observed.

Once repaired, all equipment passes through our normal pre-shipment checkout procedure. Return shipping expense on out-of-warranty repairs will be charged to the customer.

For instruments out of warranty, the customer must supply a purchase order number for the repair before the item will be returned to him.

SHIPPING DAMAGE

Shipments should be carefully examined when received for evidence of damage caused by shipping. If damage is found, immediately notify Canberra and the carrier making delivery, as the carrier is normally responsible for damage caused in shipment. Carefully preserve all documentation to establish your claim. Canberra will provide all possible assistance in processing damage claims.

Due to the delicate nature of cooled detectors [Ge(Li) and Si(Li)] Canberra requires that delivery to and from air freight terminals be handled with special care. Do not ship such Detectors without first obtaining advice from our Traffic Department.

REVISION B
July 1, 1978

ASYNCHRONOUS LINE INTERFACE
for
C.I. DATA SYSTEM MODULES

Model 1788

CONTENTS

		Page
1	INTRODUCTION	1-1
2	SPECIFICATIONS	
2.1	Inputs	2-1
2.2	Outputs	2-1
2.3	Controls/Indicators	2-1
2.4	Power	2-1
2.5	Physical	2-2
2.6	Accessories	2-2
3	CONTROLS, CONNECTORS, ADJUSTMENTS, AND INDICATORS	
3.1	Inputs	3-1
3.2	Input/Output Connectors	3-1
3.2.1	Control In and Control Out	3-1
3.2.2	EIA Connector	3-1
3.3	X Out	3-3
3.4	Front Panel Controls and Lights	3-3
3.4.1	Flag Pushbutton	3-3
3.4.2	Front Panel Light	3-3
3.5	Internal Jumpers	3-4
4	OPERATING INSTRUCTIONS	
4.1	General	4-1
4.2	Data Readout Format	4-2
5	THEORY OF OPERATION	
5.1	Operation as a Receiver (Computer to Model 1788)	5-1
5.2	Operation as a Transmitter (Model 1788 to Computer)	5-1

ASYNCHRONOUS LINE INTERFACE
for
C.I. DATA SYSTEM MODULES
Model 1788

Section 1

INTRODUCTION

The Model 1788 Interface is a character-buffered communications interface designed to assemble or disassemble serial information required by a communication device for parallel transfer of control to, or data from, the line of Canberra Industries' data acquisition modules.

The Model 1788, when combined with any of the Canberra Data Acquisition modules, will provide the user with a very powerful data system, capable of simple control and data readout via a standard EIA RS-232C serial line interface. This control and data readout can be performed using a computer directly, or via the telephone lines using a MODEM.

The unit is connected using a "Daisy Chain" closed loop to any Canberra printing instrumentation such as the Model 1776, 1774 or 1773 line of Scalers. The Model 1788 is then connected to a standard EIA port.

Section 2

SPECIFICATIONS

2.1 INPUTS

CONTROL IN	Canberra "Daisy Chain" input; rear panel
SIGNAL IN (#)	Rear panel BNC; a low-to-high or high-to-low level shift (jumper selectable) will transmit a "#" to the computer Input impedance; ≈ 3K ohms A low voltage is - 15VDC to +0.8VDC A high voltage is +3VDC to +15VDC
EXTERNAL BAUD RATE	Rear panel BNC; a TTL input for externally determining the Model 1788 Baud rate; input must be 16 times the desired baud rate

2.2 OUTPUTS

CONTROL OUT	Rear panel; Canberra "Daisy Chain" output
SIGNAL OUT (X)	Rear panel BNC, Computer generated output; internal jumpers allow for a. 4μsec Low to High pulse b. 4μsec High to Low pulse c. Level state (Low or High) corresponding to the state of the front panel indicator lamp (Will current sink 200ma in the low state)
EIA DATA	Rear panel EIA RS-232C connector

2.3 CONTROLS/INDICATORS

FLAG	Front panel pushbutton which transmits the double quote (") character to the computer
ATTENTION	Front panel indicator light controlled by the computer

2.4 POWER

+24V	–	0mA
- 24V	–	0mA
+12V	–	35mA
- 12V	–	20mA
115VAC	–	70mA

(rear panel 1/2 amp S.B. fuse and holder)

2.5 PHYSICAL

SIZE	Standard double-width NIM module (2.70 x 8.74 inches) (6.85 cm x 22.19 cm per TID 20893 (Rev)
NET WEIGHT	4.1 lb (1.86 kg.)
SHIPPING WEIGHT	9.1 lb (4.10 kg.)

2.6 ACCESSORIES

One Canberra Model 1404 Data/Control Cable, 3 ft. in length is supplied with unit

Section 3

CONTROLS, CONNECTORS, ADJUSTMENTS, AND INDICATORS

3.1 INPUTS

3.1.1 16X BAUD RATE IN

When using the unit with an External Clock Mode, a frequency of 16 times the desired Baud Rate must be used (not higher than 520kHz). The input impedance is T^2L.

3.1.2 SIGNAL IN (#)

This BNC connector will accept either a low-to-high or high-to-low level shift having any rise time and minimum duration of 100ns. Low voltage is from - 15 to +0.8 volts, high voltage is from 3 volts to 15 volts, input impedance 3K ohms. Two internal jumpers are used to select which level shift will send the computer a # (043); these jumpers are labeled on the P.C. Board as "L" and "#".

3.2 INPUT/OUTPUT CONNECTORS

3.2.1 CONTROL IN AND CONTROL OUT

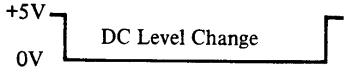
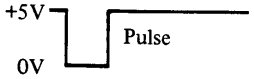
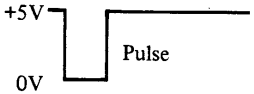
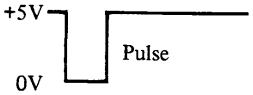
These connectors provide the system control commands of Start, Stop, Reset, Print, and BCD Data. See Figure 3-1.

3.2.2 EIA CONNTECTOR

This 25-pin connector provides the signals to meet the standard EIA format. The following signals are used when interfacing to a computer which has an RS 232-C Interface.

EIA CONNECTOR

Pin	Signal	Jumper Position	Description
1	AA	N/A	Ground
2	BB	C2 (in) M2 (out)	Received Data (from computer to 1788)
3	BA	C1 (in) M1 (out)	Transmitted Data (from 1788 to computer)
4	CA	M3 out	Not Used (tied to +5V)
5	CB	C3 in	+5 volts (Clear to Send)
6	CC	C3 in	+5 volts (Data Set Ready)
7	AB	N/A	Ground
8	CF	C3 in	+5 (Received Line Signal Detection, not used)

PIN No.	FUNCTION	SIGNAL DESCRIPTION
1	BCD Data "1"	Logic 1 = +4V, Logic 0 = 0V
2	BCD Data "2"	Logic 1 = +4V, Logic 0 = 0V
3	BCD Data "4"	Logic 1 = +4V, Logic 0 = 0V
4	BCD Data "8"	Logic 1 = +4V, Logic 0 = 0V
5*	Print Clock	
6	System Hold	
7	System Stop	
8	System Start	
9	System Reset	
15	Ground	

* Print Clock pulse train from Data Scanner will appear on Pin 5 of the CONTROL IN connector after system has finished printing its data contents.

Figure 3-1. Control IN/Out Connector Wiring

The following signals are used when connection is to a Modem.

PIN (Aux J105)	Signal	Jumper Position	Description
1	AA	N/A	Ground
2	BA	C1 (out) M1 (in)	Transmitted Data (from 1788 to Modem)
3	BB	C2 (out) M2 (in)	Received Data (from Modem to 1788)
4	CA	M3 (in)	Request to Send
5	CB	C3 out	Clear to Send
6	CC	C3 out	Data Set Ready
7	AB	N/A	Ground
8	CF	C3 out	+5 (Received Line Signal Detector)
20	CD	N/A	+5 (Data Terminal Ready)

NOTES:

A. A signal with a +5 before the description means that it is always in the spacing or "on" condition.

B. The user should get a copy of the EIA standard dated Aug. 1969 published by:

Electronic Industries Association
Engineering Department
2001 Eye Street N.W.
Washington, D.C. 20006

3.3 SIG OUT (X)

Internal Jumper N (Normal): provides a 4μsec pulse either from low-to-high or high-to-low (internal jumper X or $\overline{X}$) when the computer sends an X command. Will current sink 200mA in the low state.

Internal Jumper S (Spec. State): provides a level shift via commands A and B, thus when the front panel light is ON, X will stay in one of its two states. When the computer turns the front panel light OFF, X will change state. If jumper X is connected, SIG OUT (X) will go high if an A is received and low if a B is received. If jumper $\overline{X}$ is connected SIG OUT (X) will low if an A is received and high if a B is received.

By leaving out the XL jumper the SIG OUT (X) BNC goes to the collector of a 30V transistor; with the XL jumper in, a 470 ohm load to +5V is placed on this collector.

3.4 FRONT PANEL CONTROLS AND LIGHTS

3.4.1 FLAG PUSHBUTTON

When pushed will send out a quote (") (044) on the serial line interface.

3.4.2 FRONT PANEL LIGHT

May be turned On by sending an (A) or OFF by sending a (B) from the computer to the Serial Line Interface.

3.5 INTERNAL JUMPERS

T	Used only at Canberra.
M_1 M_2 M_3	Should be in place when going to a modem, left out when going to a computer.
C_1 C_2 C_3	Should be in place when going to a computer, left out when going to a modem.
O	Parity. The "O" position provides odd parity, the remaining position provides even parity.
P	The "P" position enables parity, the non "P" position will turn off the parity.
1S	The 1S position provides on Stop bit, the NON 1S position provides two Stop bits.
(#L), (#) and ($\overline{\#}$)	Used to set up the rear panel BND SIG in (#)
X, $\overline{X}$ and XL	Used to set up the Rear panel BNC SIG OUT (X).
Ext, 96, 48, 24, 3, 12	Use one jumper only to set the Baud Rate. (Multiply by 100)

Section 4

OPERATING INSTRUCTIONS

4.1 GENERAL

The purpose of this section is to familiarize the user with the operation and use of the Communications Interface Model 1788. This unit is used to connect any Canberra Data Acquisition modules to a computer or modem using the standard EIA communications port. The user should set up the following jumpers first:

	Jumper	Shipped from Factory
1)	Baud Rate	9600
2)	Parity	Even
3)	Parity Control	On
4)	C	C - Jumpers In
5)	M	M - Jumpers Out
6)	1S	Set for one Stop Bit
7)	#L and #	Set for Positive Input
8)	X and XL	Set for T^2L Output

Controls commands from the computer (or modem) to the Model 1788:

ASCII BASE 8	Character	Definition
122	R	Will Reset the system.
101	A	Will turn on the F.P. light.
123	S	Will start the system.
102	B	Will turn off the F.P. light.
110	H	Will halt the system.
130	X	Will give the X output pulse. (Jumper N in.)
111	I	Will cause the Data Acquisition modules to go into their Readout mode (if they are not already in the Readout mode) and the next unit in order will read out its six characters followed by a space (040).

Data and controls from the Model 1788 to the computer (or modem).

ASCII	Character	Definition
040	Space	Follows data from modules.
041	!	The system has just stopped or there are no more scalers to read.
042	"	Given from pushing the FLAG button.
043	#	Given from an input to Sig. In (#).
060	0	Data from modules
061	1	Data from modules
062	2	Data from modules
063	3	Data from modules
064	4	Data from modules
065	5	Data from modules
066	6	Data from modules
067	7	Data from modules
070	8	Data from modules
071	9	Data from modules

4.2 DATA READOUT FORMAT

When the computer is ready to readout the data system, an "I" command will be sent to the Model 1788. The first "I" will put the system in PRINT MODE and collect the data from the first scaler. The DATA FORMAT will consist of the six characters of the scaler (MSD first) followed by a space (040). The Model 1788 will then wait for either another "I" (which will take the data from the next scaler) or an "S" (which will start the data system). The Model 1788 will not transmit either CR or LF. When the Model 1788 has read out the last scaler, the next "I" will take the scalers out of PRINT MODE and transmit a "!" back to the computer.

A typical system operation format is:

STEP	COMPUTER COMMAND	1788 COMMAND
1	R →	RESET SYSTEM
2	S →	START SYSTEM
3	Wait for system done "!" from the 1788 or computer could give an "H" (in which case a "!" will still be given by the 1788)	
4	I* → Data from 1st scaler is read.	System goes in PRINT MODE.
5	I → Data from 2nd scaler is read	Request next scaler data.
6a	I → "!" (There were only two scalers in the system).	Request next scaler data.
or		
6b	S →	Unit goes out of PRINT MODE and starts counting.
7 (is 6a)	S →	Unit starts counting.

*If the scaler had 127325 as its number, it would come out as 127325 (space).

Section 5

THEORY OF OPERATION

5.1 OPERATION AS A RECEIVER (COMPUTER TO MODEL 1788)

Serial information comes into the UART I.C. (A4) on Pin 20, is converted to a 7 bit parallel word and fed to a decoder A3. This 7 bit code is strobed into A3 by A11 Pin 13. The resultant decoded command will appear as a negative pulse on one of the seven output pins of A3 (Pins 2, 3, 5, 6, 11, 13, and 14). Each command is defined in Table 5-1.

Character	Code	Pin on A3	Description
R	$(122)_8$	10	Provides a Reset Pulse on Pin 9 of the Control Out Connector (J4).
S	$(123)_8$	13	Provides a Start Pulse on Pin 8 of J4, turns OFF the Print Flip Flop A8 Pin 6, will stop the Model 1788 via A15 Pin 1.
H	$(110)_8$	5	Provides a Stop Pulse on J4 Pin 7, sets the "!" transmit Flip Flop A28 Pin 6.
X	$(130)_8$	14	Provides a pulse out of the rear panel BNC "Sig. Out (X)" (Set by jumpers).
I	$(111)_8$	6	Sets the Print Flip Flop (or Readout Flip Flop) A8 Pin 6. This will cause one six-digit unit to read out (followed by a space). The data modules will stay in the Print Mode (lights out) until either a Start "S" is given or an "I" is given. After all units have been read out, another "I" will cause a "!" to be transmitted via A22 Pin 9. The seven character printout sequence is controlled by counter A16 and Flip Flop A18.
A	$(101)_8$	2	Will set Flip Flop A8 and turn on the Front Panel light.
B	$(102)_8$	3	Will clear Flip Flop A8 turning off the Front Panel light.

Table 5-1

5.2 OPERATION AS A TRANSMITTER (MODEL 1788 TO COMPUTER)

There are four conditions that cause the Model 1788 to transmit information to the computer. These are:

1. Done, transmits a ! (41)
2. Front panel button transmits a " (42)
3. Rear panel BNC Sig. In (#) transmits a # (43)
4. Data read from the Data Acquisition modules

A counter, made up of A29 and four gates (A12 P6, A13 P6, 8, and 12), strobes each of the four transmit conditions, when any one goes true the output from its gate will go low (when the counter strobes that gate).

This will turn off the counter via A12 Pin 8 and start the transmission process through A15 Pin 8. The parallel 8 bits of data are fed to A4 Pins 26-33. For each strobe of A11 Pin 12, this data is converted to serial information and transmitted from A4 Pin 25 to the EIA back panel connector. The transmission characters are explained in Table 5-2.

Transmitted Data	Code	Description
!	041	When the Model 1788 receives a Stop command from either the computer or data modules, Flip Flop A28 Pin 6 will be set to enable the setting of the strobe counter A13 Pin 6. A28 P6 can also be set by trying to read out more data modules than there are.
#	43	When an input pulse is given to the rear panel BNC Sig. In (#), Flip Flop A28 Pin 8 will be set. The strobe counter can then set Gate A13 P8, causing the Model 1788 to transmit a # (043).
"	42	When the front panel pushbutton FLAG is pushed, Flip Flop A18 Pin 8 will set, causing a " to be transmitted.
DATA	060-071	When the Model 1788 receives a "I" Flip Flop A8 Pin 6 sets. This causes Gate A13 Pin 12 to go low (when the strobe counter gets to that Gate). The Model 1788 then transmits 7 characters of data. (XXXXXX space where X=0-9, note the 7th character is always a space.)

Table 5-2

Model 2088
Communications Interface

Features

- Interface to Printers for Readout and Computers/Calculators for control
- Fully EIA RS-232C compatible
- Baud rate selectable from 110 to 38400
- CLEAR, START, and STOP control and readout capabilities over expandable data acquisition system

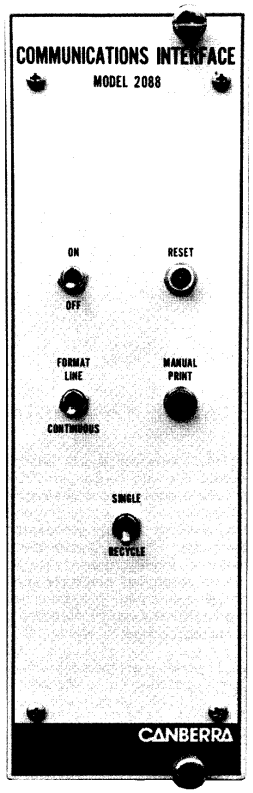
Description

The Canberra Model 2088 Communications Interface combines a Canberra "Daisy Chain" readout scanner and EIA RS-232C interface. This enables a computer to control and readout up to fifty compatible scalers and timers (such as Canberra Models 2071A and 2072A with -02 options) or permits the data stored in these compatible scalers and timers to be printed on an RS-232C compatible printer.

Commands from the computer to the Model 2088 are:

R	$(52)_{16}$	Resets modules (clears data)
S	$(53)_{16}$	Starts modules counting
CAN	$(18)_{16}$	Stops count and causes readout of data from modules
X	$(58)_{16}$	Output pulse on rear panel BNC
XON	$(11)_{16}$	In band flow control. Transmission interrupted at conclusion of current character
XOFF	$(13)_{16}$	

The Model 2088 has been designed for easy cable-connection with any unit in a Canberra Nuclear Counting System. Provision has been made for all control and interfacing required to allow direct plug-in cable assembly of the printout system.



Specifications

INPUT

CONTROL INPUT - 15-pin connector accepts PRINT (completed) Command from the last unit to be recorded, and character serial BCD information from each successive unit; Command Signal: 5 volt negative-going pulse from a +5 volt level, maximum rise time 500 nanoseconds, minimum width 2 microseconds; Data Signals: positive logic levels, where logical 1 = +4 volts, and logical 0 = 0.2 volts; dc coupled; rear panel Amphenol 17-20150 connector.

OUTPUTS

CONTROL OUTPUT - 15-pin connector provides PRINT, HOLD, RESET and START command; 5 volt negative-going pulse from a +5 volt level, maximum rise time 500 nanoseconds, minimum width 2 microseconds; HOLD command: dc level change from +5 volts to 0.5 volts maximum during printout; dc coupled; rear panel Amphenol 17-10150 connector.

SIGNAL OUTPUT (X) - Rear panel BNC; computer generated output; 4 μsec wide; low to high* or high to low (jumper selectable) TTL pulse; 470 ohm pullup to +5 V* or open collector (jumper selectable).

INPUT/OUTPUT

EIA - Provides standard RS-232C data compatible with external EIA printers; EIA flow control accepts Flag, DTR (RS-232C level) or XON/XOFF character protocol to control EIA output; character input to generate SIGNAL output pulse, Daisy Chain RESET, START or STOP/Readout; ±12 V (jumper selectable) to provide power for Model 9182 EIA to 20 mA Current Loop Adapter; rear panel 25-pin Female Dataphone-compatible connector.

CONTROLS

ON/OFF - Front panel toggle switch to enable or inhibit data recording after data accumulation.

FORMAT - Front panel toggle switch to select LINE or CONTINUOUS data print format; LINE selection causes each new data cycle record of one or more scalers, timers, or other input devices to begin on a new line of the printed output; CONTINUOUS selection causes each record to begin where the last record ended; line feed and carriage return then occurs after each tenth six-digit data field; this format is particularly useful when only one or two units are printed per data recording cycle; eight digit data field and line feed/carriage return after every eighth data field internally jumper selectable; CR $(0D)_{16}$* or ! $(21)_{16}$ (jumper selectable) transmitted at the end of record. Alternate selectable program provides a carriage return after every data field and a SPACE $(20)_{16}$ or SUB $(1A)_{16}$ at the end of record.

MANUAL PRINT - Front panel pushbutton causes entire data system to stop and data recording to begin.

RESET - Front panel pushbutton causes scanner to reset to the initial condition and to provide a carriage return and line feed signal when the next print cycle is initiated.

SINGLE/RECYCLE - Front panel toggle switch to select SINGLE or RECYCLE data accumulation by the counting system.

INTERNAL CONTROLS - DIP switch selectable Baud Rate: (110, 150, 300*, 600, 1200, 2400, 4800, 7600, 19.2k, 38.4k); Even*, Odd, or No Parity bit; One* or two Stop bits; Internal jumper selectable Modem* or Terminal Emulation; Enable or Disable* Carriage Return Delay; Printing or Non-printing* End-of-Transmission Character; positive* or negative TTL signal output; six* or eight digit data field; eight or ten* fields per line; EIA connector ±12 V power connected or disconnected.*

*Factory Setting

PERFORMANCE

CAPACITY - Will provide the capability to record up to fifty compatible data accumulation units.

OPERATING TEMPERATURE - 0 to 50°C.

CONNECTORS

CONTROL INPUT - Rear panel; 15-pin Amphenol 17-20150.

CONTROL OUTPUT - Rear panel; 15-pin Amphenol 17-10150.

EIA - Rear panel, 25 pin Female Dataphone compatible.

SIGNAL OUTPUT (X) - Rear panel; BNC UG-1094/U.

POWER REQUIREMENTS

+24 V — 0 mA +12 V — 330 mA
-24 V — 0 mA -24 V — 20 mA

PHYSICAL

SIZE - Standard dual-width module 6.85 cm (2.70 inches) per TID-20893. (Rev.)

NET WEIGHT - 1.17 kg (2.6 lbs.)

SHIPPING WEIGHT - 3.42 kg (7.6 lbs.)

ACCESSORIES

One Canberra C1404 Daisy Chain Cable is included with the 2088.

COMMUNICATIONS INTERFACE
Model 2088

0187

Operator's Manual

Canberra Industries, Inc., One State Street, Meriden, CT 06450 Tel. (203) 238-2351 TWX: 710-461-0192 Cable: CANBERRA

M194R1 3/86 Printed in U.S.A.

Table of Contents

		Page
1.	**INTRODUCTION**	
1.1	Description	1
2.	**CONTROLS AND CONNECTORS**	
2.1	Front Panel	2
2.2	Rear Panel	3
2.3	Internal Controls	4
2.4	Jumper Description	5
2.5	S1 Switch Positions	5
2.6	Control IN and Control OUT	5
2.7	EIA Connector	6
2.8	SIG OUT (X)	7
3.	**OPERATION**	
3.1	Installation	7
3.2	Printer Operation	7
3.3	Computer Operation	8
3.4	Computer Commands	8
4.	**READIN/READOUT FORMAT AND CONTROLS**	
4.1	Printer Readout Format	8
4.2	Computer Readout Format	9
4.3	External Computer Control	9
5.	**COMPUTER PROGRAMMING CONSIDERATIONS AND EXAMPLES**	
5.1	Simple Word Counting	10
5.2	FORTRAN IV Example	11
6.	**THEORY OF OPERATION**	
6.1	Block Diagram Description	12
6.2	Daisy Chain Interface	13
6.3	EIA Output Formatter	13
6.4	I/O	13
6.5	EIA Input Decode	13
6.6	Control Logic	13

		Page
APPENDIX A		
SPECIFICATIONS		
A.1	Input	14
A.2	Outputs	14
A.3	Input/Output	14
A.4	Controls	14
A.5	Performance	14
A.6	Connectors	14
A.7	Power Requirement	14
A.8	Physical	14
A.9	Accessories	14
APPENDIX B		
MODEL 207x-02 DAISY CHAIN MODIFICATION FOR 8 DIGIT READOUT		15

FIGURE LISTING

Figure 2.1	Front Panel	2
Figure 2.2	Rear Panel	3
Figure 2.3	Internal Jumpers	4
Figure 2.4	Control In/Out Connector Wiring	5
Figure 3.1	Typical Counting System	7
Figure 4.1	Three Readouts in the FORMAT-LINE Mode	8
Figure 4.2	Three Readouts in the FORMAT-CONTINUOUS Mode	9
Figure 4.3	Three Readouts in the FORMAT-LINE Mode	9
Figure 4.4	Three Readouts in the FORMAT-CONTINUOUS Mode	9
Figure 5.1	Program Flow Chart	10
Figure 6.1	Model 2088 Block Diagram	12

Section 1.
Introduction

1.1 DESCRIPTION

The Model 2088 Communications Interface is a Daisy Chain readout controller with an EIA RS-232C interface. This enables a computer to control and read out up to fifty counters and timers (such as Canberra Models 2071A and 2072A with -02 options) or permits the data stored in these units to be printed on an RS-232C compatible printer.

Various EIA formats, including baud rates from 110 to 38 400 baud, are selectable with internal DIP switches.

Computer commands to the Model 2088 via the full duplex serial RS-232C EIA interface allow control of the CLEAR, START, STOP and readout capabilities of the Canberra Daisy Chain.

Without computer control, the Model 2088 operates as a Scanner. Daisy Chain modules are read out to a printer via the rear panel EIA connector.

LINE or CONTINUOUS readout format is front panel switch selectable.

The SINGLE/RECYCLE mode switch provides a method of automatically placing the Daisy Chain system back into the counting mode after readout.

Section 2.
Controls and Connectors

2.1 FRONT PANEL

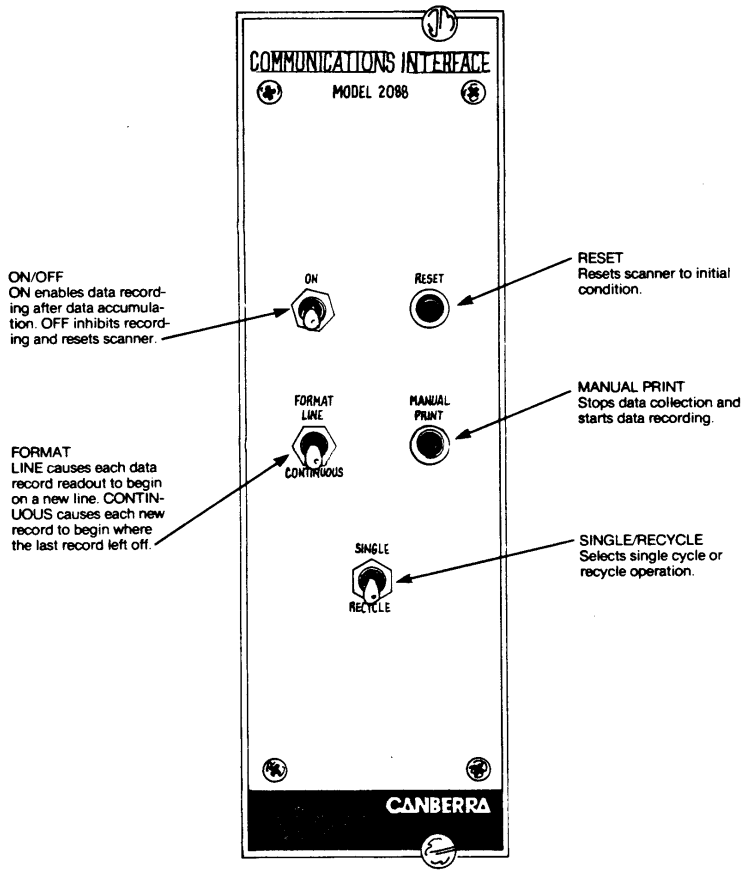


Figure 2.1
Front Panel

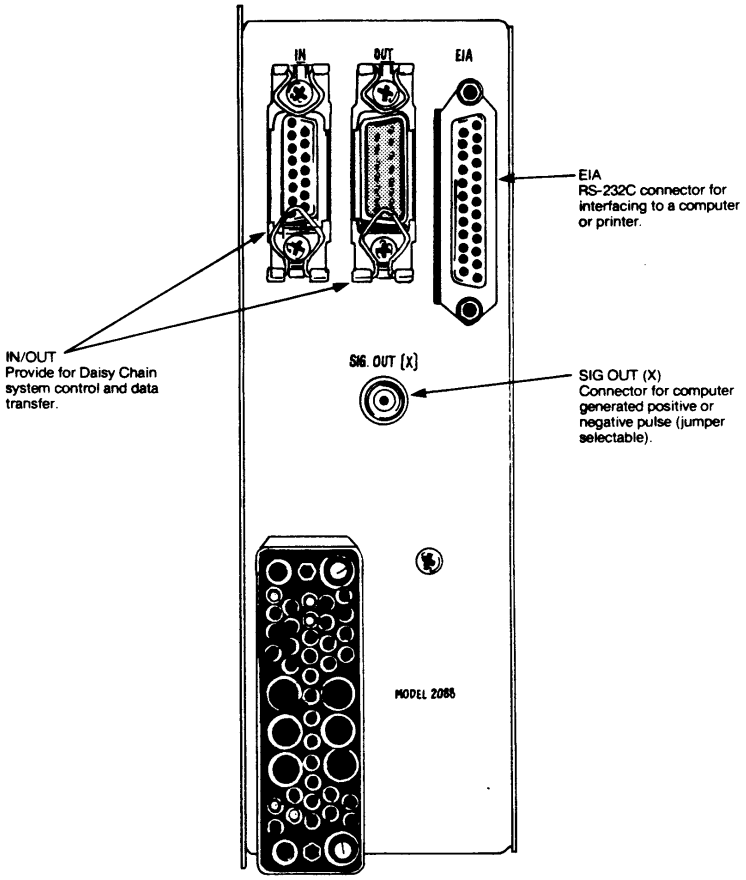


Figure 2.2
Rear Panel

2.3 INTERNAL CONTROLS

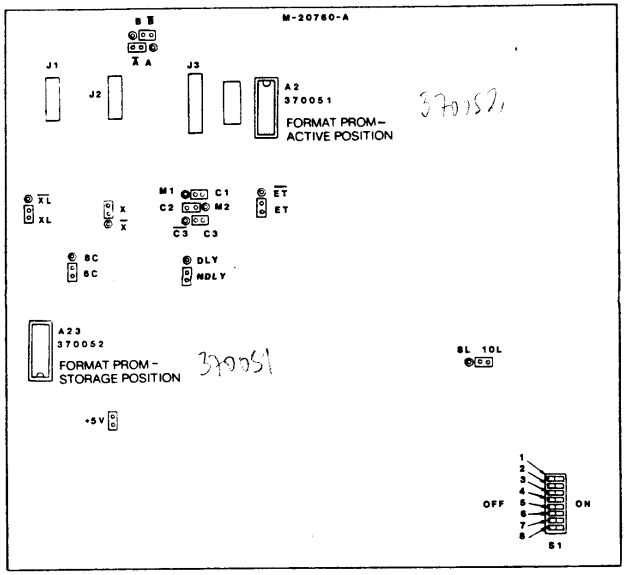


Figure 2.3
Internal Jumpers, Left Side Cover Removed

JUMPER	FUNCTION
* A/	Inactive position
A	Connects -12 V to EIA connector pin 10
* B/	Inactive position
B	Connects +12 V to EIA connector pin 9
* XL	Sig Out pulled up to + 5 V
XL/	Sig Out open collector
* 6C	Selects 6 characters per data word
8C	Selects 8 characters per data word
* X	Selects positive polarity of Sig Out
X/	Selects negative polarity of Sig Out
* C1, C2	Select standard computer/printer configuration for EIA data input and output.
M1, M2	Select terminal configuration for EIA data input and output.
* C3	Select internal pullup to EIA control lines
C3/	Select external pullup to EIA control lines

JUMPER	FUNCTION
* NDLY	Selects no delay after carriage return
DLY	Selects 160 millisecond delay after carriage return
* ET	Selects EOT termination character (SPACE with computer PROM)
ET/	Selects ! printer termination character (SUB with computer PROM)
* 10L	Selects 10 words per line
8L	Selects 8 words per line
* A2 = 370051	Readout Format PROM for Printer selected
A2 = 370052	Readout Format PROM for Computer selected
* + 5 V	Factory test jumper
S1-1 to S1-8	Selects Baud Rate, Parity, Data Bits, Stop Bits; see Section 2.4

*Factory Setting

2.4 JUMPER DESCRIPTION

- A, A/, B, B/-Model 9182 Power - Shipped in the inactive A/, B/ positions. In the A, B positions +12 and -12 V are provided to power the Model 9182 EIA to 20 mA Current Loop Adapter.

- 6C, 8C-Chars per Data Word - Shipped in the 6C, six decimal digits per Daisy Chain word, position. Move to position 8C for reading out Daisy Chain modules that are capable of transmitting eight digits per Daisy Chain word. Note that, for reading out with a Model 2088, six and eight digit modules cannot be intermixed on the daisy chain.

- 10L, 8L-Words per Line - Shipped in the 10L, ten Daisy Chain words per line, position. Move to the 8L position, eight Daisy Chain words per line, if the printer overprints.

- C3, C3/-Hardware Handshake - Shipped in the C3 position, no hardware handshake required.

- XL, XL/-Sig Out - Shipped in the XL position, 470 ohm pullup to + 5 V. Move to XL/ position for open collector output.

- X, X/-Sig Out Polarity - Shipped in the X position, positive going signal output. Move to X/ position for negative going signal output.

- A2-Readout Format - Shipped with 370051 Printer Readout Format PROM installed in location A2. For Computer operation, replace PROM in A2 with 370052 Computer Readout Format PROM (found in storage position A23).

2.5 S1 SWITCH POSITIONS

- EIA Format - Configured for 300 Baud, No Parity, Seven data bits, and Two Stop Bits. Switch S1 is used to change this format.

Table 2.1 EIA Format

S1 - PC ROCKER SW 8 POS.	
POS. - 1	ON = ODD PARITY OFF = EVEN PARITY*
POS. - 2	ON = 7 BITS* OFF = 8 BITS
POS. - 3	ON = 1 STOP BIT OFF = 2 STOP BITS*
POS. - 4	ON = PARITY ENABLED OFF = PARITY DISABLED*
POS. - 5, 6, 7 and 8 see Table 2.2	

Table 2.2 Baud Rate Chart

S1 POS. BAUD	8	7	6	5
110	OFF	OFF	OFF	ON
150	ON	OFF	OFF	ON
300*	OFF	ON	OFF	ON
600	ON	ON	OFF	ON
1200	OFF	OFF	ON	ON
2400	ON	OFF	ON	ON
4800	OFF	ON	ON	ON
9600	ON	ON	ON	ON
19.2 k	OFF	ON	ON	OFF
38.4 k	ON	ON	ON	OFF

*Factory setting.

2.6 CONTROL IN AND CONTROL OUT

These connectors provide BCD data and four system control commands: Start, Stop, Reset, and Print. See Figure 2.4.

PIN No.	FUNCTION	SIGNAL DESCRIPTION
1	BCD Data "1"	Logic 1 = +4V, Logic 0 = 0V
2	BCD Data "2"	Logic 1 = +4V, Logic 0 = 0V
3	BCD Data "4"	Logic 1 = +4V, Logic 0 = 0V
4	BCD Data "8"	Logic 1 = +4V, Logic 0 = 0V
5*	Print Clock	+5V, 0V, Pulse Train
6	System Hold	+5V, +0.5V, DC Level Change
7	System Stop	+5V, 0V, Pulse
8	System Start	+5V, 0V, Pulse
9	System Reset	+5V, 0V, Pulse
10	Display BCD "1"	Logic 1 = 0V, Logic 0 = +4V
11	Display BCD "2"	Logic 1 = 0V, Logic 0 = +4V
12	Display BCD "4"	Logic 1 = 0V, Logic 0 = +4V
13	Display BCD "8"	Logic 1 = 0V, Logic 0 = +4V
14	Display Clock	+12V, Pulse Train, +5V, 0V
15	Ground	

Figure 2.4
Control In/Out Connector Wiring

NOTES:

All data and control lines except Print Clock line (pin 5) are internally wired directly between corresponding pins of the CONTROL IN and CONTROL OUT connectors.

The Print Clock pulse train from the earlier unit in the chain will appear on pin 5 of the CONTROL IN connector after the earlier unit has finished printing its data contents.

The signals on pins 10 through 14 are not used by this unit.

2.7 EIA CONNECTOR

2.7.1 Factory Setup (Modem Emulation)

The following signals are used for interfacing to a computer or printer with an RS-232C Interface.

EIA CONNECTOR

Pin	Signal	Jumper Position	Description	Direction ref. to 2088
1	GND	—	Frame Ground	Common
2	RXD	C2 (in) M2 (out)	Received Data	IN
3	TXD	C1 (in) M1 (out)	Transmitted Data	OUT
4	RTS	—	Request to Send (15 k pullup to + 5 V)	IN
5	CTS	C3 in	Clear to Send (1 k pullup to + 5 V)	IN
6	DSR	—	Data Set Ready (1 k pullup to + 5 V)	OUT
7	GND	—	Signal Ground	Common
8	CD	C3 in	Carrier Detect (1 k pullup to + 5 V)	OUT
9	+12 V	A/	Open	—
10	- 12 V	B/	Open	—
11	Flag	C3 in	Flag line (1 k pullup to + 5V)	IN
20	DTR	C3 in	Data Terminal Ready (1 k pullup to + 5 V)	IN

NOTE: If the computer or printer pulls line 5, 11, or 20 (CTS, FLAG, or DTR) low, the Model 2088 will suspend transmission at the end of the current character.

2.7.2 Alternate RS-232C Configuration.

The following signals are used when the 2088 emulates a Terminal.

Pin	Signal	Jumper Position	Description	Direction ref. to 2088
1	GND	—	Frame Ground	Common
2	TXD	C1 (out) M1 (in)	Transmitted Data	OUT
3	RXD	C2 (out) M2 (in)	Received Data	IN
4	RTS	—	Request to Send (15 k pullup to + 5 V)	OUT
5	CTS	C3 in	Clear to Send (1 k pullup to + 5 V)	IN
6	DSR	—	Data Set Ready (1 k pullup to + 5 V)	OUT
7	GND	—	Signal Ground	Common
8	CD	C3 out	Carrier Detect	IN
9	+12 V	A/	Open	—
10	- 12 V	B/	Open	—
11	Flag	C3 out	Flag line	IN
20	DTR		Data Terminal Ready	IN

NOTES: A. Lines 5, 11, and 20 are open and must be pulled up. Installing jumper C3 or putting Line 8 to +12 V will pull these lines high, or they can be individually pulled up. If the computer or printer pulls line 5, 11, or 20 (CTS, FLAG, or DTR) low, the Model 2088 will suspend transmission at the end of the current character.

B. The EIA RS-232C standard is available from:
Electronic Industries Association, Engineering Department, 2001 Eye Street N.W., Washington, DC 20006.

2.7.3 EIA Flow Control
Transmission of data from the Model 2088 can be controlled using the FLAG/DTR handshake lines and/or using XON/XOFF protocol (see Section 4.3.5). Either method stops data transmission at the end of the current character.

2.8 SIG OUT (X)
Internal Jumper X-X/: provides a 4 μs pulse either from low-to-high (X) or high-to-low (X/) when the computer sends an X command. Will current sink 200 mA in the low state.

By leaving out the XL jumper, the SIG OUT (X) BNC goes to the collector of a 30 V transistor; with the XL jumper in, a 470 ohm load to + 5 V is placed on this collector.

Section 3. Operation

The purpose of this section is to familiarize the user with the operation and use of the Model 2088. This unit is used to connect Canberra Data Acquisition modules to a computer, modem, or printer using the standard EIA communications port.

3.1 INSTALLATION
The Canberra Model 2000 Bin and Power Supply or other bin and power supply systems conforming with the mechanical and electrical standards set by AEC Report TID-20893 (Rev.) will accommodate the Model 2088. The right side cover of the NIM module acts as a guide for insertion of the instrument. Secure the module in place by turning the two front panel captive screws clockwise until finger tight. It is recommended that the NIM bin power switch be OFF whenever the module is installed or removed.

3.2 PRINTER OPERATION
A block diagram of a typical Canberra counting system operating with a printer is shown in Figure 3.1.

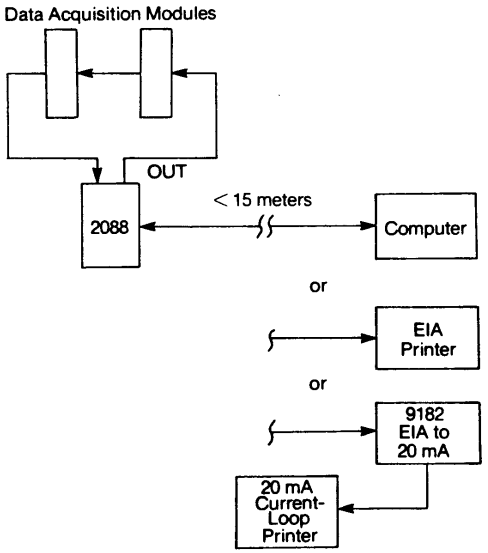


Figure 3.1
Typical Counting System

A. Prior to installation, the internal jumpers should be set to their desired position. See figure 2.3 for location of each jumper.

   Baud rate, number of Stop bits, Parity, and data length must be set, via S1, to match the printer's settings. If you are providing your own cable, refer to section 2.7.

   For Model 9182 EIA to Current Loop operation, jumpers A and B must be selected.

B. Insert the Model 2088 and the Data Acquisition Modules into a standard NIM bin. Interconnect the modules and Printer as shown in Figure 3.1.

C. Switch the Model 2088 and all Data Acquisition Modules to single cycle before applying Bin power.

D. Set the Model 2088 ON/OFF switch to ON, and the FORMAT switch to LINE.

E. Clear all Data Acquisition Modules, then press the Model 2088's MANUAL PRINT button. The printer should print each module's data channels. The module connected to the Model 2088's OUT connector will print first, followed by each succeeding module in the Daisy Chain.

   In the LINE FORMAT mode, the printouts will follow one another on the current print line, up to 8 (or 10) data words per line, depending on the position of the 8L-10L internal jumper (Figure 2.3).

   Every time MANUAL PRINT is pressed, the readout will start on a new line.

F. Change the Model 2088 SINGLE/RECYCLE switch to RECYCLE. Select a short preset time on the counter and START it. After the preset time is reached, the data should be read out as before to the printer, and the modules should go back into a count cycle. This will be repeated until the Model 2088 SINGLE/RECYCLE switch is put in the SINGLE position. Data for each cycle should be printed on a separate line.

G. Switch the FORMAT switch from LINE to CONTINUOUS. Eight or ten data words (depending on jumper 8L/10L) are now printed on each line before a carriage return and line feed are sent. The next MANUAL PRINT will start where the last printout ended rather than on a new line.

3.3 COMPUTER OPERATION

A. Prior to installation, the internal jumpers should be set to their desired position. See Figure 2.3 for location of each jumper.

   Baud rate, number of Stop bits, Parity, and data length must be set, via S1, to match the computer's settings.

   Replace standard Printer Readout Format PROM in location A2 with the Computer Readout Format PROM (370052) stored in location A23. (While the Printer Format PROM will work with a computer, programming is made much simpler with the Computer Format PROM.)

B. Insert the Model 2088 and the Data Acquisition Modules into a standard NIM bin. Interconnect the modules and computer as shown in Figure 3.1

C. Switch the Model 2088 and all Data Acquisition Modules to SINGLE before applying Bin power.

D. Set the Model 2088 ON/OFF switch to ON, and FORMAT switch to LINE.

3.4 COMPUTER COMMANDS

Computer commands to the Model 2088 via the full duplex Serial RS-232 EIA interface allow control of the CLEAR, START, STOP and readout capabilities of the Canberra Daisy Chain.

Commands from the computer to the Model 2088 are:

Char.	Key	Hex	Dec	
R	R	52	82	Resets modules (clears data)
S	S	53	83	Starts modules counting
CAN	CTL-X	18	24	Stops count and causes readout of data from modules. Readout also begins when any module reaches preset.
X	X	58	88	Output pulse on rear panel BNC
X-ON	CTL-Q	11	17	In band flow control. Transmission interrupted at conclusion of current character.
X-OFF	CTX-S	13	19	

Section 4.
Readin/Readout Format and Controls

This section defines the readout format with the Printer Readout Format PROM (number 370051) or the Computer Readout Format PROM (number 370052) installed in location A2 of the Model 2088 printed circuit board. Refer to Figure 2.3 for location.

4.1 PRINTER READOUT FORMAT

There can be either six or eight characters per Daisy Chain word depending on the setting of the 6C/8C jumper. A Daisy Chain word is defined as the data in a single scaler, timer, or counting channel. The Models 2071A and 2072A contain two 6- or 8-digit words (see Appendix B). Other modules may have the capacity to transmit six digits only. It is not possible to mix modules with 6- and 8-digit words; all words must be the same length for proper operation. A SPACE is transmitted after every word.

There can be either eight or ten Daisy Chain words per printing line depending on the setting of the 8L/10L jumper. A Carriage Return, Line Feed sequence is generated before the beginning of a new line. Ten words per line is the factory setting.

A terminating character is transmitted after the last Daisy Chain word is read out. With the jumper in the ET position this character is an EOT (4 Hex) and in the ET/ position it is a "!" (21 Hex).

```
CR,LF,
111111,SP,222222,SP,333333,SP,444444,SP,ET
CR,LF,
111111,SP,222222,SP,333333,SP,444444,SP,ET
CR,LF,
111111,SP,222222,SP,333333,SP,444444,SP,ET
```

Figure 4.1
Three Printer Readouts in the FORMAT-LINE Mode

```
CR,LF,
111111,SP,222222,SP,333333,SP,444444,SP,ET,111111,SP,222222,SP,333333,SP,444444,SP,ET,111111,SP,222222,SP,
CR,LF,
333333,SP,444444,SP,ET
```

Figure 4.2
Three Printer Readouts in the FORMAT-CONTINUOUS Mode

```
SP,SP,
111111,CR,222222,CR,333333,CR,444444,CR,ET
SP,SP,
111111,CR,222222,CR,333333,CR,444444,CR,ET
SP,SP,
111111,CR,222222,CR,333333,CR,444444,CR,ET
```

Figure 4.3
Three Computer Readouts in the FORMAT-LINE Mode

```
SP,SP,
111111,CR,222222,CR,333333,CR,444444,CR,ET,111111,CR,222222,
CR,333333,CR,444444,CR,ET,111111,CR,222222,CR,SP,SP
333333,CR,444444,CR,ET
```

Figure 4.4
Three Computer Readouts in the FORMAT-CONTINUOUS Mode

NOTES: CR = Carriage Return; LF = Line Feed; SP = Space; ET = Selected terminating character; Numbers 111111 to 444444 equate to the six digit readout of each Daisy Chain word 1 through 4; Commas (,) are shown for clarity, and are not part of the format.

4.2 COMPUTER READOUT FORMAT

There can be either six or eight characters per Daisy Chain word depending on the setting of the 6C/8C jumper. A Daisy Chain word is defined as the data in a single scaler, timer, or counting channel. The 2071A and 2072A contain two 6- or 8-digit words (see Appendix A). Other modules have the capacity to transmit six characters only. It is not possible to mix modules with 6- and 8-digit words; all words must be the same length for proper operation. A CARRIAGE RETURN is transmitted after every word.

A terminating character is transmitted after the last Daisy Chain word is read out. This character is a SPACE (20 Hex) with the ET-ET/ jumper in the ET position, and a SUB (1A Hex) with the jumper in the ET/ position.

4.3 EXTERNAL COMPUTER CONTROL

A computer can control the Daisy Chain and some internal functions of the Model 2088 via EIA data input to the Model 2088.

4.3.1 System Reset

When the computer transmits an "R" (52 Hex) the Model 2088 sends a RESET signal out on the Daisy Chain and all modules on the Daisy chain are reset (data cleared to zero). This signal does not affect the Model 2088 itself.

4.3.2 System Start

When the computer transmits an "S" (53 Hex) the Model 2088 sends a START signal out on the Daisy Chain and all modules on the Daisy Chain start counting. This signal does not affect the Model 2088 itself.

4.3.3 System Stop

When the computer transmits a "CAN" (Control X, 18 Hex) the Model 2088 sends a STOP signal out on the Daisy Chain and all modules on the Daisy Chain stop counting. The Model 2088 also generates a HOLD on the Daisy Chain, scans the Daisy Chain data, and transmits this data to the computer. If RECYCLE has been selected on the Model 2088 front panel, the 2088 then sends a RESET, START sequence on the Daisy Chain, clearing and starting the modules.

4.3.4 Signal Out

When the computer transmits an "X" (58 Hex) the Model 2088 provides a 4 μs pulse either from low-to-high or high-to-low (jumper selectable) to the Rear Panel SIG OUT (X) BNC.

4.3.5 XOFF/XON

When the computer or printer transmits an XOFF (Control S, 13 Hex) the Model 2088 is inhibited from sending out data to the computer. If the XOFF was transmitted during readout of data, the transmission is halted at the conclusion of the current character. Transmission is allowed (or continued) when the computer or printer transmits an XON (Control Q, 11 Hex). Note that the XOFF does not abort the readout. The readout must be completed, or the Daisy Chain will 'hang'.

Section 5.
Computer Programming Considerations and Examples

5.1 SIMPLE WORD COUNTING

1. Install the 370052 Computer Readout Format PROM in location A2 (Figure 2.3) of the Model 2088.
2. Select SPACE as an End of Text character (ET-ET/ jumper in the ET position).
3. Write a program following the flow chart in Figure 5.1

The first time through the loop, the program will wait at the INPUT block for the Daisy Chain to stop counting and read out. This function can be initiated by a module reaching preset, the Model 2088 front panel MANUAL PRINT button being pressed, any counting module on the Daisy Chain being manually stopped at its front panel, or by the Model 2088 receiving a CAN (Control X, 18 HEX) from the computer.

This loop assumes that the computer is able to input data fast enough without losing characters. If characters are lost, either input buffering or XOFF, XON protocol is needed. Consult your computer manual for specifics on I/O operation.

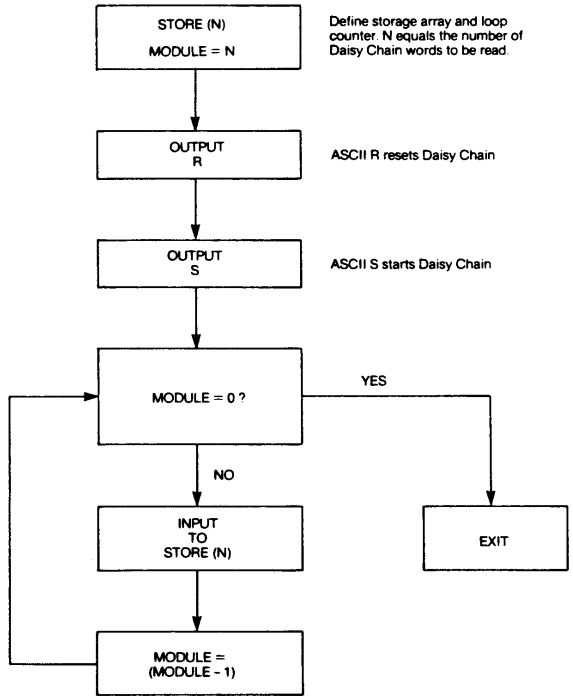


Figure 5.1
Program Flow Chart

5.2 FORTRAN IV EXAMPLE

```
ccccc
c     Test program to demonstrate a simple application for the
c     Model 2088 with the End of Transmission character
c     SUB (1A HEX) used to signal the computer that the last
c     Daisy Chain Word has been read.
c
c     The Model 2088 is set for SINGLE cycle and is configured
c     with the Computer Readout Format PROM and the ET-ET/ jumper
c     in the ET/ position.
c
c     The computer resets and starts the Daisy Chain, and readout
c     is started when a counting module in the Daisy Chain reaches
c     preset or is manually stopped.
c
c     This program was tested with a VAX*11/750 and a PDP*11.
c
c     Lun 10 was externally assigned to the terminal port connected
c     to the Model 2088 EIA output.
c
c

      integer      lun, nmods, res, sta
      integer*2    nimdat (30)              !maximum of 30 NIM data
                                            !words chosen
      lun = 10                              !establish 2088 line
      res = 82                              !R command ascii code
      sta = 83                              !S command ascii code

10    write (lun, 11) res, sta              !reset 2088; start counting
11    format ('+', 2a1, $)

      i = 1                                 !start reading data back;
15    read (lun, 20, end = 35) nimdat (i)   !record returned count,
                                            !end if SUB received
      i = i + 1
      goto 15

20    format (i8)

35    rewind lun                            !move file pointer to beg.
      nmods = i-1
      type 39 nmods
39    format (' ',i4,' words read back: ', $)   !read out header
      do 50 i = 1, nmods                    !     and
      write (5, 40) nimdat(i)               !module data
40    format ('+ ', i6, $)
50    continue

      goto 10
      end
c
c
c     End of example.
c
ccccc
```

*VAX and PDP are registered trademarks of Digital Equipment Corporation

Section 6.
Theory of Operation

The following is a description of the circuitry used in the Model 2088. Components are referred to with reference designations such as Q1 or A5. Throughout the following circuit description, refer to the circuit block diagram and schematics.

6.1 Block Diagram Description

Daisy Chain In signals enter the EIA Output Formatter where they are formatted and sent to the I/O section for transmission to an external printer or computer. Serial Data In to the I/O section is sent to the EIA Input Decode which accepts valid input control signals and routes them to either the control logic or to the Daisy Chain out. The Control Logic keeps track of the number of characters per word, and the number of words per line and initiates the transmission of spaces, carriage returns, line feeds, and End of Text characters. It also generates the clock pulses to the Daisy Chain and generates the carriage return delay.

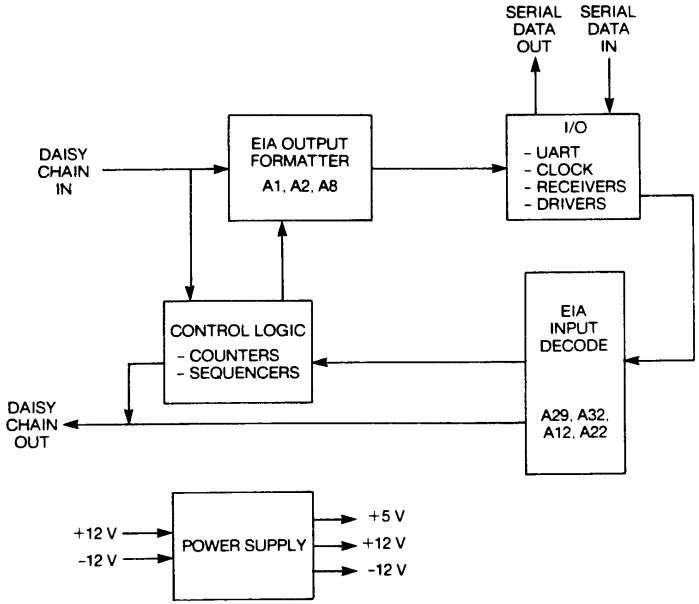


Figure 6.1
Model 2088 Block Diagram

6.2 DAISY CHAIN INTERFACE

The interface provides interconnection for five control-signals, four data-bits, and ground on two 15-pin connectors.

The control signals and their functions are:

1. START/: a bidirectional negative-going pulse swinging from + 5 V to ground with a minimum width of 1 μs. The signal initiates a counting cycle in the counter to which it is applied.
2. STOP/: a bidirectional negative-going pulse going from + 5 V to ground with a minimum width of 1.5 μs. The signal terminates a counting cycle in the counter to which it is applied, and initiates readout cycle in the scanner to which it is applied.
3. RESET/: a bidirectional negative-going pulse going from + 5 V to ground with a minimum width of 1 μs. The signal clears data in the counter to which it is applied.
4. HOLD/: a negative level at ground potential; + 5 V when false. This signal, generated by the 2088, disables the counter display-update function and enables the readout function.
5. PRINT CLOCK/: a negative-going pulse going from + 5 V to ground with a minimum width of 3 μs. The signal originates at J102-5 in the 2088 and is used to request each BCD data character for the counting module. The counting module, after receiving its required number of print clock pulses, passes succeeding ones on to the next module. This signal is the only one not wired in parallel to both 15-pin control connectors. Completion of the reading out of all modules is indicated when the transmitted clock pulse is returned to the 2088 at J101-5.

The four data bits swing between + 5 V and ground, are high true, and have 1, 2, 4, 8 weights.

Readout begins with the HOLD signal going true as a result of the STOP pulse generated by a module in the system, the computer sending a stop code to the 2088, or the 2088 front panel Manual Print button being pressed. HOLD (level) is generated by the Model 2088 and the first clock pulse is sent out on the Daisy Chain. The most significant data digit of the first counting module is put on the Daisy Chain data lines. This data is sent to the UART through the formatter and transmitted to the EIA device (printer or computer). Clock pulses are transmitted until all modules in the Daisy Chain are read out.

6.3 EIA OUTPUT FORMATTER

Daisy Chain and Control logic data are sent to the PROM A2. This PROM (Computer or Printer format) modifies the Daisy Chain data and the SP, CR, LF and ET signals from the control logic into the desired format to be transmitted to the EIA device.

6.4 I/O

Data is sent to and received from the EIA port via the UART A3 and the transmit and receive buffers/level shifters A7 and A6. The transmit and receive clocks are generated by A34 and the desired Baud rate clock is selected by A33, the 8 to 1 multiplexer. Switch S1 is used to select baud rate as well as parity and number of stop bits.

6.5 EIA INPUT DECODE

Valid EIA input codes are decoded and routed by A12. The input data is strobed through A12 via one shot A22 pin 5. This one shot also resets the UART receive register via A32 pin 12. The decoder output is used to set and clear the XON, XOFF handshake flip-flop A21 pin 5, as well as pulse the Signal Out BNC and the Daisy Chain RESET, START, and STOP lines.

6.6 CONTROL LOGIC

The remaining logic controls the timing and format of the Daisy Chain data read out to the EIA port. A pulse on the STOP line (from the front panel MANUAL PRINT, the EIA device through A12, or a counting module being stopped) sets the HOLD flip-flop A21 pin 9 via one-shot A22 pin 4.

Gate A4 and one-shots A27 and A28 control the clock cycle time and generate the Daisy Chain clock via A15 pin 1. A13 counts cycles at its clock input pin 14. When 6 or 8 cycles are completed, depending on the 6C/8C jumper position, the signal SP is asserted and a space character is transmitted. The SP signal is fed to the words-per-line counter A30 via one-shot A16 pin 13.

When 8 or 10 space characters are counted, depending on the 8L/10L jumper position, the CRSET signal at A33 pin 2 sets the Carriage Return flip-flop A18 pin 6. This causes a carriage return to be transmitted, the Line Feed flip-flop A17 pin 8 to be set, and a line feed to be transmitted.

The readout cycles continue until all units on the Daisy Chain are read out and the returning Daisy Chain clock pulse sets the EOT flip-flop at A18 pin 13. After the End of Text character is sent, the CEOT signal releases the HOLD flip-flop at A21 pin 11, completing the readout cycle. If the front panel SINGLE/RECYCLE switch is in RECYCLE, the CEOT signal resets and starts the Daisy Chain for another counting cycle.

Appendix A. Specifications

A.1 INPUT

CONTROL INPUT - 15-pin connector accepts PRINT (completed) Command from the last unit to be recorded and character serial BCD information from each successive unit. Command Signal: 5 V negative-going pulse from a + 5 V level, maximum rise time 500 ns, minimum width 2 μs. Data Signals: positive logic levels, where logical 1 = + 4 V, and logic 0 = 0.2 V; dc coupled; rear panel Amphenol 17-20150 connector.

A.2 OUTPUTS

CONTROL OUTPUT - 15-pin connector provides PRINT, HOLD, RESET, and START commands: 5 V negative-going pulse from a + 5 V level, maximum rise time 500 ns, minimum width 2 μs. HOLD command: dc level change from + 5 V to 0.5 V maximum during printout; dc coupled; rear panel Amphenol 17-10150 connector.

SIGNAL OUTPUT (X) - Rear panel BNC; computer generated output; 4 μs wide; low to high or high to low (jumper selectable) TTL pulse; 470 ohm pullup to + 5 V or open collector (jumper selectable).

A.3 INPUT/OUTPUT

EIA - Provides standard RS-232C data, compatible with external EIA printers; EIA flow control accepts Flag, DTR (RS-232C level), or XON/XOFF character protocol; character input to generate SIGNAL OUTPUT pulse; Daisy Chain RESET, START or STOP/Readout; +12 V and/or -12 V (jumper selectable) to provide power for Model 9182 EIA to 20 mA Current Loop Adapter; rear panel 25-pin female Dataphone-compatible connector. Cables are available for common printers, modems and computers.

A.4 CONTROLS

ON/OFF - Front panel toggle switch to enable or inhibit data recording after data accumulation.

FORMAT - Front panel toggle switch to select LINE or CONTINUOUS data print format; LINE selection causes each new data cycle record to print on a new line; CONTINUOUS selection causes each record to begin where the last record ended; line feed and carriage return then occurs after each tenth six-digit data field (this format is particularly useful when only one or two units are printed per data recording cycle); eight digit data field and line feed/carriage return after every eighth data field, jumper selectable; EOT (4 Hex) or !(21 Hex) (jumper selectable) transmitted at the end of record. Alternate selectable program provides a carriage return after every data field and a SPACE (20 Hex) or SUB (1A Hex) at the end of record.

MANUAL PRINT - Front panel pushbutton causes entire data system to stop and data recording to begin.

RESET - Front panel pushbutton causes scanner to reset to the initial condition and to provide a carriage return and line feed signal when the next print cycle is initiated.

SINGLE/RECYCLE - Front panel toggle switch to select SINGLE or RECYCLE data accumulation by the counting system.

INTERNAL CONTROLS — Dip switch selectable Baud Rate: (110, 150, 300, 600, 1200, 2400, 4800, 9600, 19.2k, 38.4k); Even, Odd, or No Parity bit; One or two Stop bits; 7 or 8 Data bits; Internal jumper selectable Modem or Terminal Emulation; Enable or Disable Carriage Return Delay; Printing or Non-printing End-of-Transmission Character; positive or negative TTL signal output; 6- or 8-digit data field; eight or ten fields per line; EIA connector +12 V and/or -12 V power connected or disconnected.

A.5 PERFORMANCE

CAPACITY - Can record up to fifty compatible data accumulation units.

A.6 CONNECTORS

CONTROL INPUT - Rear panel; 15-pin Amphenol 17-2150.

CONTROL OUTPUT - Rear panel; 15-pin Amphenol 17-10150.

EIA - Rear panel, 25-pin female, Dataphone compatible.

SIGNAL OUTPUT (X) - Rear panel, BNC.

A.7 POWER REQUIREMENT

+ 24 V — 0 mA	+12 V — 330 mA
- 24 V — 0 mA	-12 V — 20 mA

A.8 PHYSICAL

OPERATING TEMPERATURE - 0 to 50° C with adequate air flow.

SIZE - Standard dual-width module 6.86 × 22.13 cm (2.70 × 8.71 in.)

NET WEIGHT - 1.18 kg (2.6 lbs.)

SHIPPING WEIGHT - 3.44 kg (7.6 lbs.)

A.9 ACCESSORIES

One Canberra C1404 Daisy Chain cable is included with the 2088.

Appendix B.
Model 207x-02 Daisy Chain Modification for 8 Digit Word Readout

A Model 2071/2071A Counter Timer or 2072/2072A Dual Counter Daisy Chain Interface (Option 207x-02) can be modified to read out all eight data digits in its counting channels.

Performing this modification and moving the 6C/8C jumper on the Model 2088 to the 8C position allows the user to read out all eight digits in each counting channel to the Model 2088.

This modification disables the function of switch S1 on the Daisy Chain Interface board (Opt 207x-02). This modification must be removed to use this switch to control reading out of the six most significant or six least significant digits.

The modification to the 2071/2072-02 option can be done as follows:

1. Remove the right side cover of the 2071/2072.
2. Disconnect the ribbon cable that connects the -02 option to the rear panel Daisy Chain connectors (J4).
3. Take out the four screws holding the -02 option board to the 2071/2072 and remove the -02 board.
4. Take out the four screws holding the -02 option board to the 2071/2072 and remove the -02 board.
5. On the solder side of the -02 option board, cut the artwork between A10 pin 4 and A11 pin 8.
6. Jumper A11 pin 8 and A11 pin 9 together.
7. Cut and remove or unsolder and lift A6 pin 10 from the artwork. Note that this may necessitate replacing this IC if the modification is ever reversed.
8. Jumper A9 pin 11 to A9 pin 2.
9. With the interface board's four mounting holes lined up with the four standoffs, press the board gently down to mate the interboard connectors.
10. Fasten the interface board to the standoffs with the four screws that were removed.
11. Align the ribbon cable's pins with the sockets of J4 at the rear of the interface board and press the pins firmly downinto the connector until flush with the tops of the connector.
12. Replace the side cover.

BASIC WARRANTY

Equipment manufactured by Canberra Industries, Inc. is warranteed against defects in materials and workmanship for a period of twelve months from date of shipment, provided that the equipment has been used in a proper manner as detailed in the instruction manuals. During the warranty period, repairs or replacement will be made at Canberra's option on a return to factory basis. The transportation cost, including insurance to and from Canberra, is the responsibility of the Customer except for defects discovered within 30 days after receipt of equipment where shipping expense will be paid by Canberra to and from Canberra.

The customer must obtain an authorized customer service return number before returning any equipment to the Canberra factory. *Compliance with this provision by the customer shall be a condition of this warranty.* In giving shipping instructions, Canberra shall not be deemed to have assumed any responsibility or liability in connection with the shipment.

The Canberra Basic Warranty applies only to equipment manufactured by Canberra which is returned to the factory. If equipment must be repaired at the customer's site, the actual repair labor and parts will be provided at no charge during the warranty period. However, travel expenses to and from the customer's site, (travel time labor, and living expenses while on site), shall be paid by the customer unless an On-Site Warranty Option has been purchased. This option may only be purchased prior to shipment of the equipment to the customer.

The express warranties set forth herein are the only warranties with respect to the products, or any materials or components purchased from others and furnished by Canberra, and there are no other warranties, expressed or implied. The warranty of merchantibility is expressly limited as herein provided and all warranties of fitness are expressly disclaimed and excluded. Canberra shall have no liability for any special, indirect or consequential damages, whether from loss of production or otherwise, arising from any breach of warranty hereunder or defect or failure of any product or products sold hereunder.

EXCLUSIONS
Warranty service is contingent upon the proper use of all equipment and does not cover equipment which has been modified without Canberra's written approval or which has been subjected to unusual physical or electrical stress as determined by Canberra Service personnel. Canberra Industries shall be under no obligation to furnish warranty service (preventive or remedial): (1) if adjustment, repair or parts replacement is required because of accident, neglect, misuse, failure of electrical power, air conditioning, humidity control, transportation, or causes other than ordinary use; (2) if the equipment is maintained or repaired or if attempts to repair or service equipment are made by other than Canberra personnel without the prior approval of Canberra.

This warranty does not cover detector damage caused by warm-up or by neutrons or heavy charged particles. Damage from these causes is readily identifiable as described in the manual accompanying each detector.

Although Canberra may frequently supply, as part of systems, equipment manufactured by other companies, the only warranty that shall apply to such non-Canberra equipment is that warranty offered by the original manufacturer, if any.

Canberra will, upon request, offer, as an option, warranty coverage for non-Canberra equipment such as computers and peripherals sold as part of a system supplied by Canberra. Quotations on this coverage may be obtained by contacting Canberra Customer Service or any of our sales staff.

SOFTWARE
Canberra warrants software media from defects discovered within 30 days after receipt.

Canberra assumes no responsibility for user-written programs or programs published as part of information exchange in Canberra periodicals.

Engineering assistance for software development is available and can be contracted through the Sales Department.

INSTALLATION
Installation of equipment purchased from Canberra shall be the sole responsibility of the customer unless the installation is specifically contracted for at the prevailing Canberra field service rates. To insure timely installation after receipt of equipment, it is recommended that installation be contracted for at the time the equipment is ordered.

ON-SITE WARRANTY OPTION
The On-Site Warranty Option provides for free on-site warranty work (Canberra pays all travel and living expenses) within the first 90 days after delivery of equipment to the customer. If installation is ordered from Canberra, the 90 day period commences upon completion of the initial installation. After the 90 day period, labor and materials used on site will still be covered by the basic warranty, but the customer shall pay for all travel expenses—travel time labor and living expenses incurred for any on-site service.

A maintenance contract may be purchased covering the period after the 90 days on-site warranty period, or after initial installation of the equipment. This is to be contracted through Canberra Customer Service.

REPAIRS
Any Canberra-manufactured instrument no longer in its warranty period may be returned, freight prepaid, to our factory for repair and realignment. When returning instruments for repair, contact the Customer Service Department for shipping instructions and an Authorized Customer Service Return Number.

All correspondence concerning repairs should include the Model number and a description of the problem observed.

Once repaired, all equipment passes through our normal preshipment checkout procedure. Return shipping expense on out-of-warranty repairs will be charged to the customer.

For instruments out of warranty, the customer must supply a purchase order number for the repair before the item will be returned.

SHIPPING DAMAGE
Shipments should be carefully examined when received for evidence of damage caused by shipping. If damage is found, immediately notify Canberra and the carrier making delivery, as the carrier is normally responsible for damage caused in shipment. Carefully preserve all documentation to establish your claim. Canberra will provide all possible assistance in processing damage claims.

Due to the delicate nature of cooled detectors (Ge and Si(Li)), Canberra requires that delivery to and from air freight terminals be handled with special care. Do not ship such Detectors without first obtaining advice from our Traffic Department.

RETURN SHIPMENTS
Canberra Customer Service Department must be notified in advance if equipment is to be returned for any reason. Canberra can suggest the best means of shipping and will be able to expedite the shipment in case it is lost or delayed in transit.

The customer must obtain an authorized customer service return number before returning any equipment to the Canberra factory. *Compliance with this provision by the customer shall be a condition of this warranty.* In giving shipping instructions, Canberra shall not be deemed to have assumed any responsibility or liability in connection with the shipment.

Equipment should be returned to your area service center or to Canberra, Meriden. For shipment from outside the U.S., our shipping address is:

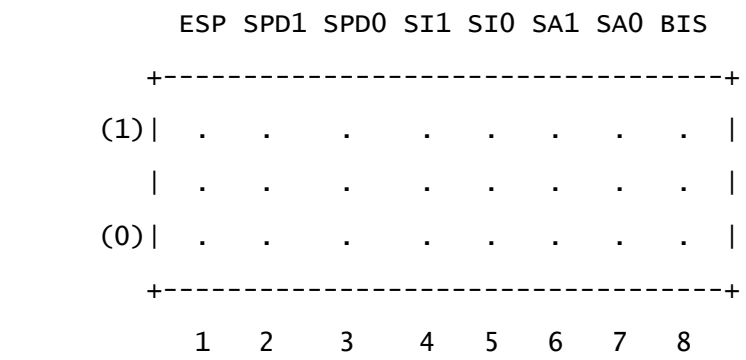
Canberra Industries, Inc.
c/o EMO-TRANS, Inc.
Bradley International Airport
Windsor Locks, CT 06096 U.S.A.

SERVICE AND SERVICEABILITY
Canberra has gone to great lengths to insure that the instruments provided are functionally modular and therefore easy to service. In addition to modularity, Canberra has embarked on an extensive System Service Program to provide a totally responsive service capability. Complete Service Contracts with special arrangements for 24 hour response and weekend standby services are available from Canberra. For a detailed description of our Customer Service Program, please contact our Systems Service Department in Meriden, Connecticut, U.S.A.

Rev. 5/1/86

ZyXEL I/O-card "SP111AT" jumper settings

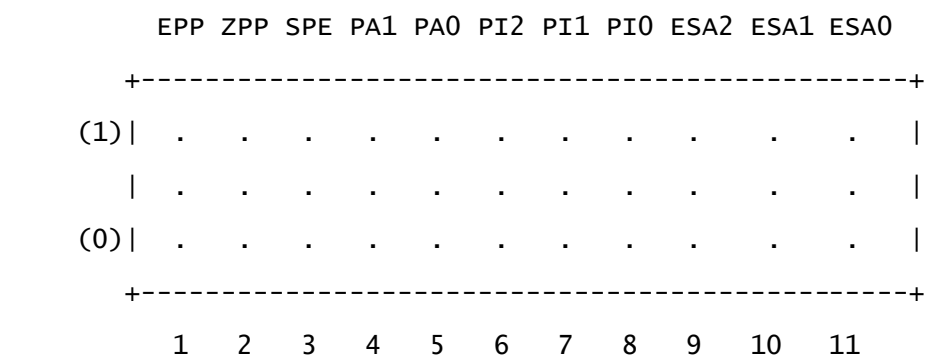
Jumper JP1:



JP1 [1]	ESP		"Enable Serial Port"
	(1)/open:		enabled
	(0)		disabled
JP1 [2+3]	SPD1	SPD0	"Speed" Multiplier for DTE rate
	(0)	(0)	1x
	(0)	(1)	2x
	(1)	(0)	4x
	(1)	(1)	8x
JP1 [4+5]	SI1	SI0	"Serial port Interrupt"
	(0)	(0)	IRQ9
	(0)	(1)	IRQ3
	(1)	(0)	IRQ4
	(1)	(1)	IRQ5
JP1 [6+7]	SA1	SA0	"Serial port Address"
	(0)	(0)	COM1 0x3F8
	(0)	(1)	COM2 0x2F8
	(1)	(0)	COM3 0x3E8
	(1)	(1)	COM4 0x2E8
JP1 [8]	BIS		n/a

(1) default setting

Jumper JP2:



JP2 [1]	EPP		"Enable Parallel Port"
	(1)/open		enabled
	(0)		disabled
JP2 [2+3]	ZPP	SPE	"ZyXEL Parallel Port" and "Serial Port Emulation"
	(0)	-	normal standard parallel port mode
	(1)	(0)	special ZyXEL Parallel Port mode
	(1)	(1)	Serial Port Emulation mode
JP2 [4+5]	PA1	PA0	"Port Address" for parallel port
	(0)	(0)	0x378 LPT1
	(0)	(1)	0x278 LPT2
	(1)	(0)	0x338 -
	(1)	(1)	0x238 -

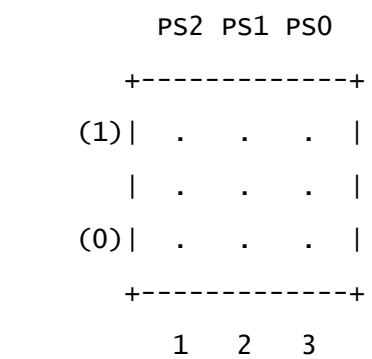
JP2 [6+7+8]	PI2	PI1	PI0	"Port Interrupt" for parallel port
	(0)	(0)	(0)	IRQ9
	(0)	(0)	(1)	IRQ3
	(0)	(1)	(0)	IRQ4
	(0)	(1)	(1)	IRQ5
	(1)	(0)	(0)	IRQ7

(1)	(0)	(1)	IRQ10
(1)	(1)	(0)	IRQ11
(1)	(1)	(1)	IRQx

note: when "IRQx" is selected, the IRQ is determined by the jumper JP5. That JP5 has four positions labeld IRQ14, IRQ15, IRQ12 and IRQ6. Depending on the jumper position, one of the labeld IRQs will be selected!

JP2 [9+10+11]	ESA2	ESA1	ESA0	"Emulated Serial Address"	
	(0)	(0)	(0)	0x3F8	COM1
	(0)	(0)	(1)	0x2F8	COM2
	(0)	(1)	(0)	0x3E8	COM3
	(0)	(1)	(1)	0x2E8	COM4
	(1)	(0)	(0)	0x3D8	-
	(1)	(0)	(1)	0x2D8	-
	(1)	(1)	(0)	0x3C8	-
	(1)	(1)	(1)	0x2C8	-

Jumper JP3:

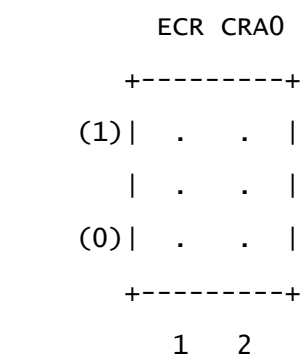


JP3 [1+2+3]	PS2	PS1	PS0	"Parallel port cycle Speed selection"
	(0)	(1)	(1)	fast speed cycle
	(0)	(0)	(1)	
	(0)	(1)	(0)	

(0) (0) (0) mid speed cycle
(1) (1) (1)
(1) (0) (1)
(1) (1) (0)
(1) (0) (0) slow speed cycle (factory default)

note: Jumper block JP3 determines the cycle time data transfer accross the parallel port connection in special "ZyXEL parallel port" (ZPP) mode and in "serial emulated port" (SPE) operation mode (see jumpers JP2[2+3] !)

Jumper JP4:

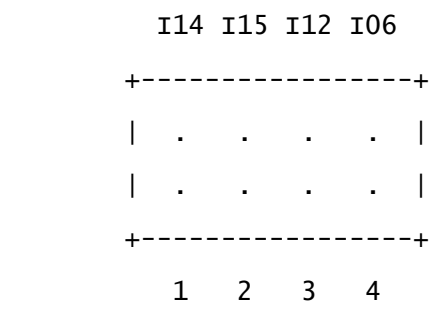


JP5 [1] ECR "Enable Control Registers"
(1)/open enabled
(0) disabled

JP4 [2] CRA0 "Control Register base Address"
(1) 0x3E0
(0) 0x2E0
note: Control Register CR0 is located on that base address
Control Register CR1 is base address +1 (eg: 3E1 or 2E1)

control Register CR2 is base address +2 (eg: 3E2 or 2E2)

Jumper JP5:



JP5	IRQx selection	IRQ14	IRQ15	IRQ12	IRQ6
JP5 [1]	14	set	-	-	-
JP5 [2]	15	-	set	-	-
JP5 [3]	12	-	-	set	-
JP5 [4]	6	-	-	-	set

note:
this jumper block setting is valid only if
the jumpers of JP2 [6+7+8] are set to position
(1) (which means "IRQx" selection active)

NAME

termio – general terminal interface

DESCRIPTION

All of the asynchronous communications ports use the same general interface, no matter what hardware is involved. The remainder of this section discusses the common features of this interface.

Opening a Terminal File

When a terminal file is opened, it normally causes the process to wait until a connection is established. In practice, users' programs seldom open these files; they are opened by *getty* and become a user's standard input, output, and error files. When a process group leader without a controlling terminal opens a terminal that is not already a controlling terminal, that terminal becomes the controlling terminal for that process and the terminal's distinguished process group (*tty group ID*) is set to the process group of that process. The control terminal plays a special role in handling QUIT and INTERRUPT signals, as discussed below. It is also used in handling the process group control signals. The control terminal is inherited by a child process during a *fork*(2). A process can break this association by changing its process group using *setpgrp*(2). When the process group leader that acquired the controlling terminal terminates, the distinguished process group of the controlling terminal is set to zero (indicating no distinguished process group). This allows the terminal to be acquired as a controlling terminal by a new process group leader.

If the O_NDELAY bit (see *open*(2)) is clear, an *open* will block until the type of modem connection requested (see *modem*(7)) is completed. If the O_NDELAY bit is set, an *open* will succeed and return immediately without waiting for the modem connection requested to complete.

Reading Characters

A terminal associated with one of these files ordinarily operates in full-duplex mode. Characters may be typed at any time, even while output is occurring, and are only lost when the system's character input buffers become completely full, which is rare, or when the user has accumulated the maximum allowed number of input characters that have not yet been read by some program. This limit is dependent on the particular implementation, but is at least 256. When the input limit is reached, all the saved characters are thrown away without notice.

Canonical Mode Input Processing (Erase and Kill Processing)

Normally, terminal input is processed in units of lines. A line is delimited by a new-line (ASCII LF) character, an end-of-file (ASCII EOT) character, or an end-of-line character. This means that a program attempting to read will be suspended until an entire line has been typed. Also, no matter how many characters are requested in the read call, at most one line will be returned. It is not, however, necessary to read a whole line at once; any number of characters may be requested in a read, even one, without losing information.

During input, erase and kill processing is normally done. By default, the character # erases the last character typed, except that it will not erase beyond the beginning of the line. By default, the character **kills (deletes) the entire input line,** and optionally outputs a new-line character. Both these characters operate on a key-stroke basis, independently of any backspacing or tabbing that may have been done. Both the erase and kill characters may be entered literally by preceding them with the escape character (\). In this case the escape character is not read. The erase and kill characters may be changed.

Process Group Control During I/O

For those drivers that support process group control, if a process is not in the distinguished process group of its control terminal, and both the process group of the process and the control terminal are non-zero, the process is said to be a *background process*. Any attempts by a background process to read from its controlling terminal will cause the process group to be sent a SIGTTIN signal, which will normally cause the members of that process group to stop. If,

however, a process is ignoring or holding the SIGTTIN signal, or (on systems that implement *vfork* separately from *fork*) has made a call to *vfork*(2) but not yet made a call to *exec*(2), then the process is instead returned an EIO error and no signal is sent to any process.

It is frequently undesirable for background processes to write to its controlling terminal. If the LTOSTOP bit is set, a background process is prohibited from writing to that terminal. Attempts to write cause the process group to be sent a SIGTTOU signal, which normally causes the members of the process group to stop. If the LTOSTOP bit is not set, the process is ignoring or holding SIGTTOU signals, or is in the middle of the *vfork*(2) window described above, and the process is allowed to write to the terminal. Certain ioctl system calls that set tty parameters are treated in this same fashion, except that the setting of the LTOSTOP bit is ignored. The effect is identical to that of terminal writes when the LTOSTOP bit is set.

Non-blocking Reads (O_NDELAY Flag)

If the O_NDELAY bit (see *open*(2) or *fcntl*(2)) is clear, the read request will complete whenever data is available. If the O_NDELAY bit is set, then the read request will complete, without blocking, in one of three ways: (1) If there is enough data available to satisfy the entire request, the read will complete successfully, having read all of the data, and return the number of bytes read; (2) If there is not enough data available to satisfy the entire request, the read will complete successfully, having read as much data as possible, and return the number of bytes it was able to read; (3) If there is no data available, the read will complete successfully, having read no data, and return a count of 0.

Special Characters

Certain characters have special functions on input. These functions and their default character values are summarized as follows:

INTR — (Rubout or ASCII DEL) generates an *interrupt* signal, which is sent to all processes with the associated control terminal. Normally, each such process is forced to terminate, but arrangements may be made either to ignore the signal or to receive a trap to an agreed-upon location; see *signal*(2).

QUIT — (Control-| or ASCII FS) generates a *quit* signal. Its treatment is identical to the interrupt signal except that, unless a receiving process has made other arrangements, it will not only be terminated but a core image file (called **core**) will be created in the current working directory if the implementation supports core files.

SWTCH — (ASCII NUL) is only used by the shell layers facility *shl*(1). The shell layers facility is not part of the general terminal interface. No special functions are performed by the general terminal interface when SWTCH characters are encountered.

ERASE — (#) erases the preceding character. It will not erase beyond the start of a line, as delimited by a NL, EOF, or EOL character.

KILL — (@) deletes the entire line, as delimited by a NL, EOF, or EOL character.

EOF — (Control-d or ASCII EOT) may be used to generate an end-of-file from a terminal. When received, all the characters waiting to be read are immediately passed to the program, without waiting for a new-line, and the EOF is discarded. Thus, if there are no characters waiting, which is to say the EOF occurred at the beginning of a line, zero characters will be passed back, which is the standard end-of-file indication.

NL — (ASCII LF) is the normal line delimiter. It cannot be changed or escaped.

EOL — (ASCII NUL) is an additional line delimiter, like NL. It is not normally used.

STOP (Control-s or ASCII DC3) can be used to temporarily suspend output. It is useful with CRT terminals to prevent output from disappearing before it can be read. While output is suspended, STOP characters are ignored and not read.

START (Control-q or ASCII DC1) is used to resume output that has been suspended by a STOP character. While output is not suspended, START characters are ignored and not read. The start/stop characters cannot be changed or escaped.

The character values for INTR, QUIT, ERASE, KILL, EOF, SWTCH, and EOL may be changed to suit individual tastes. The ERASE, KILL, and EOF characters may be escaped by a preceding \ character, in which case no special function is done.

Writing Characters

When one or more characters are written, they are transmitted to the terminal as soon as previously-written characters have finished typing. Input characters are echoed by putting them in the output queue as they arrive. If a process produces characters more rapidly than they can be typed, it will be suspended when its output queue exceeds some limit. When the queue has drained down to some threshold, the program is resumed.

Closing a Terminal File

When the last process to have a terminal device file open closes the file, the output data is drained before the close returns.

Modem Disconnect

When a modem disconnect is detected, a *hang-up* signal is sent to all processes that have this terminal as the control terminal. Unless other arrangements have been made, this signal causes the processes to terminate. If the hang-up signal is ignored, any subsequent read returns with an end-of-file indication. Thus, programs that read a terminal and test for end-of-file can terminate appropriately when hung up on.

Termio Structure

Several *ioctl*(2) system calls apply to terminal files. The primary calls use the following structure, defined in **<termio.h>**:

```
#define NCC    8
struct    termio {
    unsigned short    c_iflag;      /* input modes */
    unsigned short    c_oflag;      /* output modes */
    unsigned short    c_cflag;      /* control modes */
    unsigned short    c_lflag;      /* local modes */
    char              c_line;       /* line discipline */
    unsigned char     c_cc[NCC];    /* control chars */
};
```

The special control characters are defined by the array *c_cc*. The relative positions and initial values for each function are as follows:

```
0   VINTR   DEL
1   VQUIT   FS
2   VERASE  #
3   VKILL   @
4   VEOF    EOT
5   VEOL    NUL
6   reserved
7   VSWTCH  NUL
```

Input Modes

The *c_iflag* field describes the basic terminal input control:

IGNBRK	0000001	Ignore break condition.
BRKINT	0000002	Signal interrupt on break.
IGNPAR	0000004	Ignore characters with parity errors.
PARMRK	0000010	Mark parity errors.
INPCK	0000020	Enable input parity check.
ISTRIP	0000040	Strip character.
INLCR	0000100	Map NL to CR on input.
IGNCR	0000200	Ignore CR.
ICRNL	0000400	Map CR to NL on input.
IUCLC	0001000	Map uppercase to lowercase on input.
IXON	0002000	Enable start/stop output control.
IXANY	0004000	Enable any character to restart output.
IXOFF	0010000	Enable start/stop input control.
IENQAK	0020000	Enable output pacing control.

If IGNBRK is set, the break condition (a character framing error with data all zeros) is ignored, that is, not put on the input queue and therefore not read by any process. Otherwise if BRKINT is set, the break condition will generate an interrupt signal and flush both the input and output queues. If IGNPAR is set, characters with other framing and parity errors are ignored.

If PARMRK is set, a character with a framing or parity error that is not ignored is read as the three-character sequence: 0377, 0, X, where X is the data of the character received in error. To avoid ambiguity in this case, if ISTRIP is not set, a valid character of 0377 is read as 0377, 0377. If PARMRK is not set, a framing or parity error that is not ignored is read as the character NUL (0).

If INPCK is set, input parity checking is enabled. If INPCK is not set, input parity checking is disabled. This allows output parity generation without input parity errors.

If ISTRIP is set, valid input characters are first stripped to 7-bits, otherwise all 8-bits are processed.

If INLCR is set, a received NL character is translated into a CR character. If IGNCR is set, a received CR character is ignored (not read). Otherwise if ICRNL is set, a received CR character is translated into a NL character.

If IUCLC is set, a received uppercase alphabetic character is translated into the corresponding lowercase character.

If IXON is set, start/stop output control is enabled. A received STOP character will suspend output and a received START character will restart output. All start/stop characters are ignored and not read. If IXANY is set, any input character will restart output that has been suspended.

If IXOFF is set, start/stop input control is enabled. When the number of characters in the input queue exceeds a system defined value (high water mark), the system will transmit a STOP character. This should cause the attached device to stop transmitting data before the maximum allowed input has been reached. When enough characters have been read from the input queue that the number of remaining characters is less than another system defined value (low water mark), then the system will transmit a START character to cause input to be resumed. In order to avoid potential deadlock, the IXOFF bit is ignored in canonical mode whenever there is no line delimiter in the input buffer. In this case, the STOP character is not sent at the high water mark, but will be transmitted later if a delimiter is received. If all complete lines are read from the input queue leaving only a partial line with no line delimiter, the START character will be sent even though the number of characters may not be less than the low water mark. When ICANON is set and the input stream contains more characters between line delimiters than the high water mark allows, it is not guaranteed that IXOFF will prevent buffer overflow and data loss, since the STOP character may not be sent in time or at all.

If IENQAK is set, the system will transmit ASCII ENQ after every 80 characters sent and then wait until the terminal responds with ASCII ACK. The terminal will respond in this way when it has sufficiently emptied its buffer. If the terminal does not respond after 5 seconds, the system will resume transmission anyway. The ASCII ACK that the terminal sends will not get entered into the input queue if it was sent in response to ASCII ENQ.

The initial input control value is all-bits-clear.

Output Modes

The *c_oflag* field specifies the system treatment of output:

OPOST	0000001	Postprocess output.
OLCUC	0000002	Map lowercase to uppercase on output.
ONLCR	0000004	Map NL to CR-NL on output.
OCRNL	0000010	Map CR to NL on output.
ONOCR	0000020	No CR output at column 0.
ONLRET	0000040	NL performs CR function.
OFILL	0000100	Use fill characters for delay.
OFDEL	0000200	Fill is DEL, else NUL.
NLDLY	0000400	Select new-line delays:
NL0	0	
NL1	0000400	
CRDLY	0003000	Select carriage-return delays:
CR0	0	
CR1	0001000	
CR2	0002000	
CR3	0003000	
TABDLY	0014000	Select horizontal-tab delays:
TAB0	0	
TAB1	0004000	
TAB2	0010000	
TAB3	0014000	Expand tabs to spaces.
BSDLY	0020000	Select backspace delays:
BS0	0	
BS1	0020000	
VTDLY	0040000	Select vertical-tab delays:
VT0	0	
VT1	0040000	
FFDLY	0100000	Select form-feed delays:
FF0	0	
FF1	0100000	

If OPOST is set, output characters are post-processed as indicated by the remaining flags, otherwise characters are transmitted without change.

If OLCUC is set, a lowercase alphabetic character is transmitted as the corresponding uppercase character. This function is often used in conjunction with IUCLC.

If ONLCR is set, the NL character is transmitted as the CR-NL character pair. If OCRNL is set, the CR character is transmitted as the NL character. If ONOCR is set, no CR character is transmitted when at column 0 (first position). If ONLRET is set, the NL character is assumed to do the carriage-return function; the column pointer will be set to 0 and the delays specified for CR will be used. Otherwise the NL character is assumed to do just the line-feed function; the column pointer will remain unchanged. The column pointer is also set to 0 if the CR character is actually transmitted.

The delay bits specify how long transmission stops to allow for mechanical or other movement when certain characters are sent to the terminal. In all cases a value of 0 indicates no delay. If OFILL is set, fill characters will be transmitted for delay instead of a timed delay. This is useful for high baud rate terminals, which need only a minimal delay. If OFDEL is set, the fill character is DEL, otherwise NUL.

If a form-feed or vertical-tab delay is specified, it lasts for about 2 seconds.

New-line delay lasts about 0.10 seconds. If ONLRET is set, the carriage-return delays are used instead of the new-line delays. If OFILL is set, two fill characters will be transmitted.

Carriage-return delay type 1 is dependent on the current column position, type 2 is about 0.10 seconds, and type 3 is about 0.15 seconds. If OFILL is set, delay type 1 transmits two fill characters, and type 2, four fill characters.

Horizontal-tab delay type 1 is dependent on the current column position. Type 2 is about 0.10 seconds. Type 3 specifies that tabs are to be expanded into spaces. If OFILL is set, two fill characters will be transmitted for any delay.

Backspace delay lasts about 0.05 seconds. If OFILL is set, one fill character will be transmitted.

The actual delays depend on line speed and system load.

The initial output control value is all bits clear.

Control Modes

The *c_cflag* field describes the hardware control of the terminal:

CBAUD	0000037	Baud rate:
B0	0	Hang up
B50	0000001	50 baud
B75	0000002	75 baud
B110	0000003	110 baud
B134	0000004	134.5 baud
B150	0000005	150 baud
B200	0000006	200 baud
B300	0000007	300 baud
B600	0000010	600 baud
B900	0000011	900 baud
B1200	0000012	1200 baud
B1800	0000013	1800 baud
B2400	0000014	2400 baud
B3600	0000015	3600 baud
B4800	0000016	4800 baud
B7200	0000017	7200 baud
B9600	0000020	9600 baud
B19200	0000021	19200 baud
B38400	0000022	38400 baud
EXTA	0000036	External A
EXTB	0000037	External B
CSIZE	0000140	Character size:
CS5	0	5 bits
CS6	0000040	6 bits
CS7	0000100	7 bits
CS8	0000140	8 bits
CSTOPB	0000200	Send two stop bits, else one.
CREAD	0000400	Enable receiver.
PARENB	0001000	Parity enable.

PARODD	0002000	Odd parity, else even.
HUPCL	0004000	Hang up on last close.
CLOCAL	0010000	Local line, else dial-up.
LOBLK	0020000	Reserved for use by *shl*(1).

The CBAUD bits specify the baud rate. The zero baud rate, B0, is used to hang up the connection. If B0 is specified, the modem control lines (see *modem*(7)) will cease to be asserted. Normally, this will disconnect the line. For any particular hardware, impossible speed changes are ignored.

The CSIZE bits specify the character size in bits for both transmission and reception. This size does not include the parity bit, if any. If CSTOPB is set, two stop bits are used, otherwise one stop bit. For example, at 110 baud, two stops bits are required.

If PARENB is set, parity generation and detection is enabled and a parity bit is added to each character. If parity is enabled, the PARODD flag specifies odd parity if set, otherwise even parity is used.

If CREAD is set, the receiver is enabled. Otherwise no characters will be received.

The specific effects of the HUPCL and CLOCAL bits depend on the mode and type of the modem control in effect. See *modem*(7) for the details.

If HUPCL is set, the modem control lines for the port will be disconnected when the last process with the port open closes it or terminates.

If CLOCAL is set, a connection does not depend on the state of the modem status lines.

If the CLOCAL bit has been set, an *open* will return immediately without waiting for the connection. For those files on which the connection has not been established or has been lost, and for which the CLOCAL bit is not set, both *read* and *write* will return a zero character count. For *read*, this is equivalent to an end-of-file condition.

The initial hardware control value after open is B300, CS8, CREAD, HUPCL.

LOBLK is used by the shell layers facility (see *shl*(1)). The shell layers facility is not part of the general terminal interface, and the LOBLK bit is not examined by the general terminal interface.

Local Modes

The *c_lflag* field of the argument structure is used by the line discipline to control terminal functions. The basic line discipline (0) provides the following:

ISIG	0000001	Enable signals.
ICANON	0000002	Canonical input (erase and kill processing).
XCASE	0000004	Canonical upper/lower presentation.
ECHO	0000010	Enable echo.
ECHOE	0000020	Echo erase character as BS-SP-BS.
ECHOK	0000040	Echo NL after kill character.
ECHONL	0000100	Echo NL.
NOFLSH	0000200	Disable flush after interrupt, quit, or suspend.

If ISIG is set, each input character is checked against the special control characters INTR, QUIT, and the suspend characters (see Process Group Control below). If an input character matches one of these control characters, the function associated with that character is performed. If ISIG is not set, no checking is done. Thus these special input functions are possible only if ISIG is set. These functions may be disabled individually by changing the value of the control character to an unlikely or impossible value (e.g., 0377).

If ICANON is set, canonical processing is enabled. This enables the erase and kill edit functions, and the assembly of input characters into lines delimited by NL, EOF, and EOL. If ICANON is not set, read requests are satisfied directly from the input queue. A read will not be satisfied until at

least MIN characters have been received or the timeout value TIME has expired between characters. (See **Non-canonical Mode Input Processing (MIN/TIME Interaction)** below). This allows fast bursts of input to be read efficiently while still allowing single character input. The MIN and TIME values are stored in the position for the EOF and EOL characters, respectively. The time value represents tenths of seconds.

If XCASE is set, and if ICANON is set, an uppercase letter is accepted on input by preceding it with a \ character, and is output preceded by a \ character. In this mode, the following escape sequences are generated on output and accepted on input:

for:	*use*:
`	\'
|	\!
~	\^
{	\(
}	\)
\	\\

For example, **A** is input as **\a**, **\n** as **\\n**, and **\N** as **\\\n**.

If ECHO is set, characters are echoed as received.

When ICANON is set, the following echo functions are possible. If ECHO and ECHOE are set, the erase character is echoed as ASCII BS SP BS, which will clear the last character from a CRT screen. If ECHOE is set and ECHO is not set, the erase character is echoed as ASCII SP BS. If ECHOK is set, the NL character will be echoed after the kill character to emphasize that the line will be deleted. Note that an escape character preceding the erase or kill character removes any special function. If ECHONL is set, the NL character will be echoed even if ECHO is not set. This is useful for terminals set to local echo (so-called half duplex). Unless escaped, the EOF character is not echoed. Because EOT is the default EOF character, this prevents terminals that respond to EOT from hanging up.

If NOFLSH is set, the normal flush of the input and output queues associated with the quit, interrupt, and suspend characters will not be done.

The initial line-discipline control value is all bits clear.

Unless otherwise noted for a specific *ioctl* command, the *ioctl*s are restricted from use by background processes. An attempt to issue an *ioctl* from a background process will cause the process to block and may cause a SIGTTOU signal to be sent to the process group.

Non-canonical Mode Input Processing (MIN/TIME Interaction)

The MIN and TIME values are stored in c_cc[VMIN] and c_cc[VTIME], respectively, and are used when ICANON is clear. MIN represents the minimum number of characters that should be received when the read is satisfied (i.e., the characters are returned to the user). TIME is a timer of 0.10 second granularity that is used to timeout burst and short term data transmissions. The four possible values for MIN and TIME and their interactions are described below.

A. MIN > 0, TIME > 0

In this case TIME serves as an intercharacter timer and is activated after the first character is received. Since it is an intercharacter timer, it is reset after a character is received. The interaction between MIN and TIME is as follows: as soon as one character is received, the intercharacter timer is started. If MIN characters are received before the intercharacter timer expires (remember that the timer is reset upon receipt of each character), the read is satisfied. If the timer expires before MIN characters are received, the characters received to that point are returned to the user. Note that if TIME expires, at least one character will be returned because the timer would not have been enabled unless a character was received. In this case (MIN > 0, TIME > 0) the read

will sleep until the MIN and TIME mechanisms are activated by the receipt of the first character.

B. MIN > 0, TIME = 0

In this case, since the value of TIME is zero, the timer plays no role and only MIN is significant. A pending read is not satisfied until MIN characters are received after any previous read completes (i.e., the pending read will sleep until MIN characters are received). A program that uses this case to handle record-based terminal I/O may block.

C. MIN = 0, TIME > 0

In this case, since MIN = 0, TIME no longer represents an intercharacter timer. It now serves as a read timer that is activated as soon as the read system call is processed. A read is satisfied as soon as a single character is received or the read timer expires. Note that in this case, if the timer expires, no character will be returned. If the timer does not expire, the only way the read can be satisfied is if a character is received. Note that in this case the read will not sleep indefinitely waiting for a character. If no character is received within TIME * 0.10 seconds after the read is initiated, the read will return with zero characters.

D. MIN = 0, TIME = 0

In this case return is immediate. If characters are present, they will be returned to the user.

Some points to note about MIN and TIME:

1. In the above explanations one may notice that the interactions of MIN and TIME are not symmetric. For example, when MIN > 0 and TIME = 0, TIME has no effect. However, in the opposite case where MIN = 0 and TIME > 0, both MIN and TIME play a role in that MIN is satisfied with the receipt of a single character.
2. Also note that in case A (MIN > 0, TIME > 0), TIME represents an intercharacter timer while in case C (MIN = 0, TIME > 0), TIME represents a read timer.

These two points highlight the dual purpose of the MIN/TIME feature. Cases A and B, where MIN > 0, exist to handle burst mode activity (e.g., file transfer programs) where a program would like to process at least MIN characters at a time. In case A, the intercharacter timer is activated by a user as a safety measure while in case B it is turned off.

Cases C and D exist to handle single character timed transfers. These cases are readily adaptable to screen based applications that need to know if a character is present in the input queue before refreshing the screen. In case C the read is timed, while in case D it is not.

Another important note is that MIN is always just a minimum. It does not denote a record length. That is, if a program does a read of 20 bytes and MIN is 10 and 25 characters are present, 20 characters will be returned to the user. If the program had requested all characters, then all 25 characters would be returned to the user.

Primary TERMIO IOCTL Commands

The primary *ioctl*(2) system calls have the form:

ioctl (fildes, command, arg)
struct termio *arg;

The commands using this form are:

TCGETA — Get the parameters associated with the terminal and store in the *termio* structure referenced by *arg*. This command is allowed from a background process; however, the information may be subsequently changed by a foreground process.

TCSETA Set the parameters associated with the terminal from the structure referenced by *arg*. The change is immediate.

TCSETAW Wait for the output to drain before setting the new parameters. This form should be used when changing parameters that will affect output.

TCSETAF Wait for the output to drain, then flush the input queue and set the new parameters.

Additional *ioctl*(2) calls have the form:

ioctl (fildes, command, arg)
int arg;

The commands using this form are:

TCSBRK Wait for the output to drain. If *arg* is 0, then send a break (zero bits for at least 0.25 seconds).

TCXONC Start/stop control. If *arg* is 0, suspend output; if 1, restart suspended output.

TCFLSH If *arg* is 0, flush the input queue; if 1, flush the output queue; if 2, flush both the input and output queues.

The following command has the form:

ioctl (fildes, command, arg)
long *arg;

FIONREAD Returns in the long integer whose address is *arg* the number of characters immediately readable from the terminal device file. This command is allowed from a background process; however, the data itself may not be read from a background process.

System Asynchronous I/O

The following commands have the form:

ioctl (fildes, command, arg)
int *arg;

FIOSSAIOSTAT If the integer whose address is *arg* is non-zero, system asynchronous I/O is enabled. That is, enable SIGIO to be sent to the process currently designated with FIOSSAIOOWN (see below) whenever the terminal device file status changes from "no read data available" to "read data available". If no process has been designated with FIOSSAIOOWN, then enable SIGIO to be sent to the first process to open the terminal device file.

If the designated process has exited, the SIGIO signal will not be sent to any process.

If the integer whose address is *arg* is 0, system asynchronous I/O is disabled.

FIOGSAIOSTAT The integer whose address is *arg* is set to 1, if system asynchronous I/O is enabled. Otherwise, the integer whose address is *arg* is set to 0.

FIOSSAIOOWN Set process ID to receive the SIGIO signals with system asynchronous I/O to the value of the integer whose address is *arg*. The super-user may designate that any process receive the SIGIO signals. If the request is not made by the super-user, the calling process is only

allowed to designate that itself or another process whose real or saved effective user ID matches its real or effective user ID, or a process that is a descendant of the calling process, receive the SIGIO signals. If no process can be found corresponding to that specified by the integer whose address is *arg*, the call will fail, with **errno** set to ESRCH. If the request is not made by the super-user, and the calling process attempts to designate a process other than itself or another process whose real or saved effective user ID matches its real or effective user ID, or a process that is not a descendant of the calling process, the call will fail, with **errno** set to EPERM.

If the designated process subsequently exits, the SIGIO signal will not be sent to any process.

The default on open of a terminal device file is that the process performing the open is set to receive the SIGIO signals.

FIOGSAIOOWN The integer whose address is *arg* is set to the process ID designated to receive SIGIO signals.

Non-blocking I/O

FIOSNBIO

If the integer whose address is *arg* is non-zero, non-blocking I/O is enabled. That is, subsequent reads and writes to the terminal device file will be handled in a non-blocking manner (see below). If the integer whose address is *arg* is 0, non-blocking I/O is disabled.

For reads, non-blocking I/O will prevent all read requests to that device file from blocking, whether the requests succeed or fail. Such a read request will complete in one of three ways: (1) If there is enough data available to satisfy the entire request, the read will complete successfully, having read all of the data, and return the number of bytes read; (2) If there is not enough data available to satisfy the entire request, the read will complete successfully, having read as much data as possible, and return the number of bytes it was able to read; (3) If there is no data available, the read will fail and **errno** will be set to EWOULDBLOCK.

For writes, non-blocking I/O will prevent all write requests to that device file from blocking, whether the requests succeed or fail. Such a write request will complete in one of three ways: (1) If there is enough space available in the system to buffer all the data, the write will complete successfully, having written out all of the data, and return the number of bytes written; (2) If there is not enough space in the buffer to write out the entire request, the write will complete successfully, having written as much data as possible, and return the number of bytes it was able to write; (3) If there is no space in the buffer, the write will fail and **errno** will be set to EWOULDBLOCK.

To prohibit non-blocking I/O from interfering with the O_NDELAY flag (see *open*(2) and *fcntl*(2)), the functionality of O_NDELAY always supercedes the functionality of non-blocking I/O. This means that if O_NDELAY is set, the driver will perform read requests in accordance with the definition of O_NDELAY. When O_NDELAY is not set, the definition of non-blocking I/O applies.

The default on open of a terminal device file is that non-blocking I/O is disabled.

FIOGNBIO

The integer whose address is *arg* is set to 1, if non-blocking I/O is enabled. Otherwise, the integer whose address is *arg* is set to 0.

Process Group Control

Some implementations support process group control. The following structure, used with process group control, is defined in **<bsdtty.h>**:

```
struct  ltchars         {
        unsigned char   t_suspc;    /* stop process character */
        unsigned char   t_dsuspc;   /* delayed stop process character */
        unsigned char   t_rprntc;   /* reserved; must be 0377 */
        unsigned char   t_flushc;   /* reserved; must be 0377 */
        unsigned char   t_werasc;   /* reserved; must be 0377 */
        unsigned char   t_lnextc;   /* reserved; must be 0377 */
};
```

The initial value for all these characters is 0377, which causes them to be disabled. The meaning for each character is as follows:

t_suspc is used to suspend the currently active process group. A *suspend* signal (SIGTSTP) is sent to all processses in the currently active process group. Normally, each process is forced to stop, but arrangements may be made either to ignore the signal or to receive a trap to an agreed-upon location; see *signal*(2). When enabled, the typical value for this character is control-z or ASCII SUB.

t_dsuspc functions like **t_suspc**, except that the *suspend* signal is sent when a process reads the character rather than when it is typed. When enabled, the typical value for this character is control-y or ASCII EM.

Attempts to set any of the reserved characters to a value other than 0377 will cause the *ioctl* to be returned with an EINVAL error code and no change will occur.

The *ioctl*(2) system calls that use the above structure have the form:

ioctl (fildes, command, arg)
struct ltchars *arg;

The commands using this form are:

TIOCGLTC Get the process group control characters and store in the *ltchars* structure whose address is *arg*. This command is allowed from a background process; however, the information may be subsequently changed by a foreground process.

TIOCSLTC Set the process group control characters from the structure whose address is *arg*.

Additional process group control *ioctl*(2) commands have the form:

ioctl (fildes, command, arg)
unsigned int *arg;

The commands using this form are:

TIOCGPGRP Returns in the integer whose address is *arg* the currently active process group associated with the terminal. This command is allowed from a background process; however, the information may be subsequently changed by a foreground process.

TIOCSPGRP Sets the currently active process group associated with the terminal to be the one whose address is *arg*.

The following condition must be met for the request to succeed; otherwise, the error [EINVAL] is returned:

The new process group value must be in the range of valid process group ID values, or it must be zero ("no process group").

One or more of the following conditions must be met for the request to succeed; otherwise, the error [EPERM] is returned:

The effective user ID of the current process is super-user.

The process group ID whose address is *arg* matches the saved process group ID of the calling process.

If any processes exist with a process ID or process group ID that is the same as the process group whose address is *arg*, those processes must have the same real or saved user ID as the real or effective user ID of the calling process, or be descendants of the calling process.

TIOCLGET Get the process group control mode word and store in the int pointed to by *arg*. This command is allowed from a background process; however, the information may be subsequently changed by a foreground process.

TIOCLSET Set the process group control mode word from the int pointed to by *arg*.

TIOCLBIS Use the int pointed to by *arg* as a mask of bits to set in the process group control mode word.

TIOCLBIC Use the int pointed to by *arg* as a mask of bits to clear in the process group control mode word.

The following bit is defined in the process group control mode word:

LTOSTOP Send SIGTTOU for background write.

If the LTOSTOP bit is set, an attempt by a process that is not in the distinguished process group to write data causes a SIGTTOU signal to be sent to that process group.

WARNINGS

Because various HP-UX implementations use non-serial interfaces that look like terminals (e.g., internal CRT's) or "smart cards," which cannot implement the capabilities described above exactly, not all the systems can exactly meet the standard stated above. Each implementation is required to state any deviations from the standard as part of its system specific documentation.

FIOSSAIOSTAT is similar to 4.2 BSD FIOASYNC, with the addition of provisions for security. FIOGSAIOSTAT is of HP origin, complements FIOSSAIOSTAT, and allows saving and restoring system asynchronous I/O TTY state for BSD style job control. FIOSSAIOOWN is similar to 4.2 BSD FIOSETOWN, with additional provisions for security. FIOGSAIOOWN is similar to 4.2 BSD FIOGETOWN. Note also the difference that the 4.2 BSD version of this functionality used process groups, while the HP-UX version only uses processes. FIOSNBIO is the same as 4.2 BSD FIONBIO, except that it does not interfere with the AT&T O_NDELAY *open* and *fcntl* flag. FIOGNBIO is of HP origin, complements FIOSNBIO, and allows saving and restoring non-blocking I/O TTY state for BSD style job control.

DEPENDENCIES

Series 300

Data loss may occur with HP 98626/98644 serial interfaces if the effective combined data rate for all installed serial interfaces exceeds 2400 baud (for example, two interfaces running at 1200 baud and a third at 300 baud is equivalent to 2700 baud combined).

The *c_iflag* field parameter IXANY (enable any character to restart output) is not supported by the HP 98628B interface card.

The *c_iflag* field parameter IENQAK (enable output pacing control) is not supported.

Timed delays are not supported.

The HP 98628B interface does not support the following baud rates: 900, 7200, 38 400.

The FIONREAD *ioctl* command is not supported.

The *c_iflag* field parameter XCASE is not supported.

The features described under **System Asynchronous I/O, Non-blocking I/O,** and **Process Group Control** are not supported.

For the purposes of **Process Group Control During I/O,** all processes are considered to be foreground processes at all times.

Series 800

The ENQ/ACK protocol and the IENQAK bit are not supported.

Timed output delays are not directly supported. If used, an appropriate number of fill characters (based on the current baud rate) is output. The total time to output the fill characters is at least as long as the time requested.

The baud rate of 38400 is not supported by the RS-232 interface.

The system specified input flow control values are as follows: low water mark is 60, high water mark is 180, and maximum allowed input is 512.

Shell layers is not supported. Therefore, neither the special character **SWTCH** nor the control mode **LOBLK** is supported.

AUTHOR

Termio was developed by HP, AT&T, and the University of California,

FILES

/dev/console
/dev/tty*

SEE ALSO

mknod(1M), shl(1), stty(1), fork(2), ioctl(2), setpgrp(2), signal(2), stty(2), blmode(3C), sttyV6(7), tty(7), modem(7).

Model 1407R
Noise Pulser

Features

- Random time distribution tail pulse output
- Average count rate 5Hz to 250kHz
- Selectable monoenergetic output 2.5mV to 250mV

Description

The Canberra Model 1407R Noise Pulser has been designed to provide a source of constant amplitude, variable frequency, random pulses, for checking linear amplification performance from the preamplifier through the analyzer system.

The requirements for testing nuclear spectroscopy systems have become more sophisticated; consequently the effects to be tested generally include resolution and system noise, gain stability at varying pulse rates, baseline and zero shifts due to count rate, and pile-up effects at varying count rates, for different amplifier shaping time constants.

Nuclear spectroscopy systems can be tested for performance either with live sources or with pulse generators. Live systems are often difficult to test because of problems in simply and predictably establishing a desired count rate, the confusion that results from random energies as well as random pulse spacing impinging upon the detector, and the inability to obtain a high enough count rate due to the poor efficiency of Ge(Li) detectors.

Pulse generator tests are normally inadequate because of the low rates obtainable from mercury tail pulsers (60 - 90Hz), the unsuitability of the pulse shapes available from higher rate pulsers, and the difficulties found in obtaining random pulse spacing from traditional pulse generators.

The Model 1407R overcomes these limitations, as it offers selectable random count rate, up to 250kHz monoenergetic output, and tail pulse output shape. The Canberra Model 1407R features constant output amplitude, selectable via front panel attenuator controls from a full scale of 25mV to 250mV. A ten-turn potentiometer permits variable 10-1 range control of the output amplitude below these limits.

The output pulses from the Noise Pulser occur randomly in time — triggered by the noise from an internal component. The **average** rate may be selected by a ten-turn front panel potentiometer from 5Hz to more than 250kHz (typical). The linearity of this control is on the order of 2 - 5% of full scale between 15kHz and 250kHz.

Two outputs are provided. The TRIGGER OUT provides logic level pulses to drive a scaler or ratemeter. The OUTPUT connector provides positive tail pulses with a rise time of 100 nanoseconds and a fall time constant of 50 microseconds. This output is suitable for the Test Input of a charge-sensitive preamplifier, or for the input on all commercially available main amplifiers.

Specifications

OUTPUTS

OUTPUT - provides positive constant amplitude, variable frequency random tail pulses; amplitude range +2.5mV to 250mV; rise time 100 nanoseconds; fall time constant 50 microseconds; output impedance 93 ohms; DC coupled; front panel BNC connector

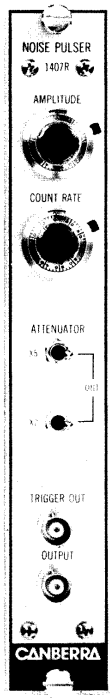
TRIGGER OUT - provides positive logic pulses; amplitude +4V (terminated in 93 ohms); rise time 100 nanoseconds, fall time 50 nanoseconds, width at half height ≈125 nanoseconds; output impedance 50 ohms; DC coupled; front panel BNC connector

CONTROLS

AMPLITUDE - front panel ten-turn potentiometer to select output pulse amplitude from 25mV to 250mV (unterminated); both Attenuator switches in OUT position

COUNT RATE - front panel ten-turn potentiometer to select average output rate from less than 5Hz to greater than 240kHz (guaranteed)

ATTENUATOR - two front panel toggle switches to select X2 and X5 attenuation of the normal 250mV full range output; total attenuation available X10



PERFORMANCE

OUTPUT - provides an essentially monoenergetic spectral line (less than 2keV FWHM) at any given amplitude or count rate

AVERAGE OUTPUT RATE - continuously variable from less than 5Hz to more than 240kHz (guaranteed); count rate linearity 2 to 5% of full scale between 15kHz and 250kHz

STABILITY - no long-term or temperature stability specifications are tested or implied

CONNECTORS

TRIGGER OUT, OUTPUT - front panel, BNC.

POWER REQUIREMENTS

+24VDC	—	40mA	+12VDC	—	25mA
-24VDC	—	25mA	-12VDC	—	40mA

PHYSICAL

SIZE - standard single-width NIM module (1.35 x 8.71 inches) 3.43 cm x 22.13 cm) per TID-20893 (rev. A).

NET WEIGHT - 1.6 lbs. (0.7 kgs.).

SHIPPING WEIGHT - 4.0 lbs. (1.8 kgs.).

RANDOM NOISE PULSER

Model 1407R

Instruction Manual

CANBERRA INDUSTRIES, INC.
45 Gracey Avenue
Meriden, Connecticut 06450

Tel: 203-238-2351

1/75

RANDOM NOISE PULSER
Model 1407R

CONTENTS

Page

1 INTRODUCTION 1-1

2 SPECIFICATIONS

2.1 Output 2-1
2.2 Performance 2-1
2.3 Controls 2-1
2.4 Connectors 2-2
2.5 Power 2-2
2.6 Physical 2-2

3 CONTROLS/CONNECTORS

3.1 General 3-1
3.2 Controls 3-1
3.3 Connectors 3-1

4 OPERATING INSTRUCTIONS

4.1 General 4-1
4.2 Initial Operation 4-1
4.3 Initial Checkout 4-1
4.4 Typical Applications 4-1

5 CIRCUIT DESCRIPTION 5-1

RANDOM NOISE PULSER
Model 1407R

Section 1

INTRODUCTION

The Canberra Model 1407R Noise Pulser has been designed to provide a source of constant amplitude, random pulses with a variable average frequency, for checking linear amplification system count rate performance from preamplifier through analyzer.

The requirements for testing nuclear spectroscopy systems have grown; the effects to be tested generally include resolution and system noise, gain stability at varying pulse rates, baseline and zero shifts due to count rate, and pileup effects at varying count rates, for different amplifier shaping time constants.

Nuclear spectroscopy systems can be tested for performance either with live sources or with pulse generators. Live systems are often not easy to test because of the inability to simply and predictably establish a desired count rate, the confusion that results from random energies as well as random pulse spacing impinging upon the detector, or the inablility to obtain a high count rate due to the poor efficiency of Ge(Li) detectors.

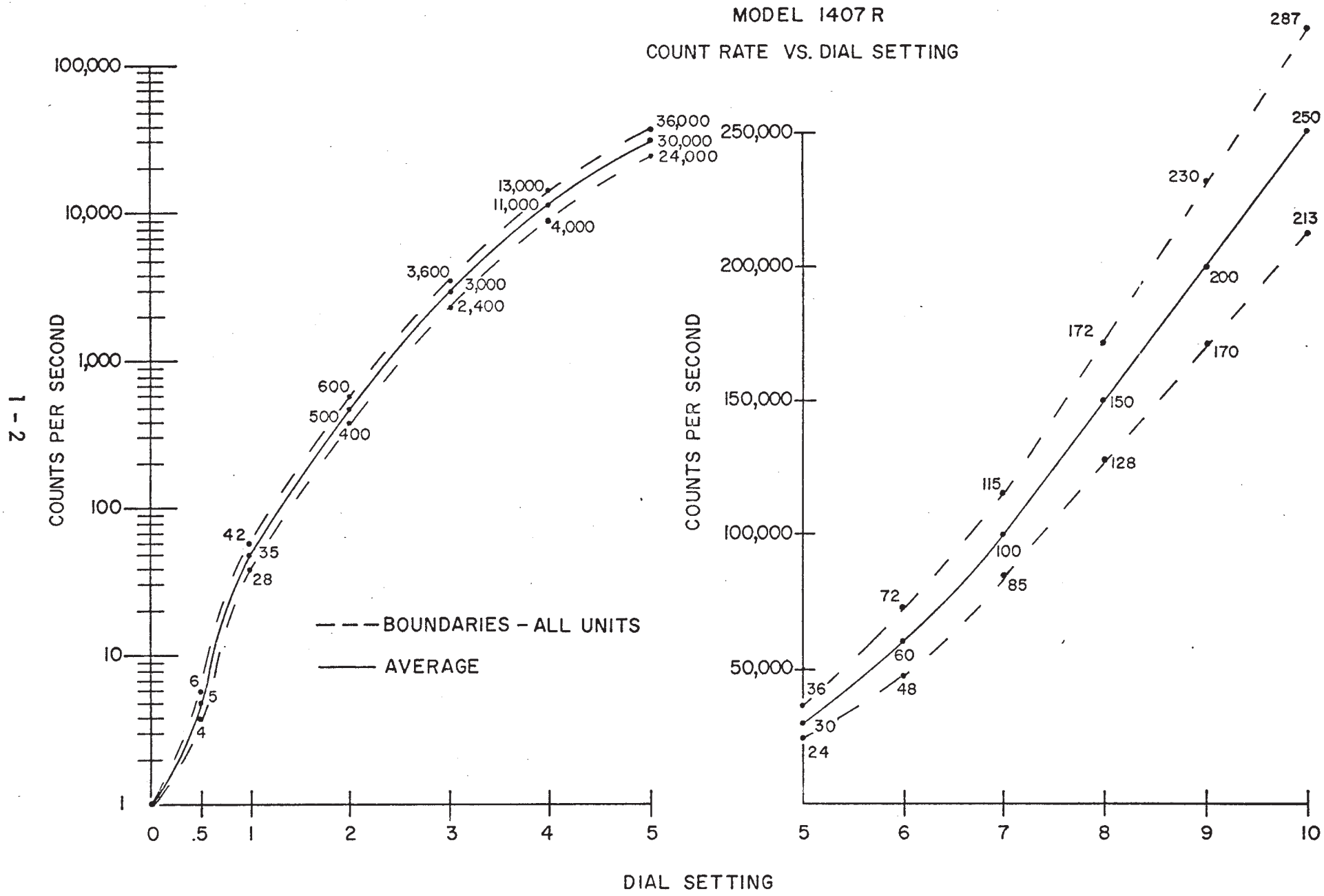
Pulse generator tests are normally inadequate because of the low rates (60-90 Hz) from mercury tail pulsers, the unsuitability of the pulse shapes available from high rate pulsers, and the difficulties found in obtaining random pulse spacing from traditional pulse generators.

The Model 1407R overcomes these limitations, as it offers selectable random count rate, up to 250,000 counts/second monoenergetic output, and tail pulse output shape. The Canberra Model 1407R features constant output amplitude, selectable via front panel attenuator controls, from a full scale of 25 mV to 250 mV. A ten-turn potentiometer permits smooth control of the output amplitude between these limits.

The output pulses from the Noise Pulser occur randomly in time, triggered by the noise from an internal noise-generating resistor. The average rate may be selected by a ten-turn front panel potentiometer from five counts/second to more than 250,000 counts/second (typical). The linearity of this control is on the order 2-5% of full scale between 15,000 counts/second and 250,000 counts/second. Lower average rates may be obtained by direct measurement, or by using the typical setting vs. rate chart given in Figure 1-1.

Two outputs are provided. The TRIGGER OUT provides logic pulses to drive a scaler or ratemeter, or to trigger an oscilloscope. The OUTPUT connector provides positive tail pulses with a rise time of 100 nanoseconds and a fall time constant of 50 microseconds. This output is suitable for the Test Input of a charge-sensitive preamplifier, or for the input on all commercially available main amplifiers.

> NOTE: If the Model 1407R is fed into the Test Input of a preamplifier, the output from the main amplifier will have an undershoot composed of two time constants, only one of which can be simply pole/zero cancelled. Thus, high rate performance of the system will suffer from the uncompensated undershoot.


MODEL 1407 R
COUNT RATE VS. DIAL SETTING
COUNTS PER SECOND
100,000
10,000
1,000
100
10
1
0
.5
1
2
3
4
5
36,000
30,000
24,000
13,000
11,000
4,000
3,600
3,000
2,400
600
500
400
42
35
28
6
5
4
BOUNDARIES – ALL UNITS
AVERAGE
250,000
200,000
150,000
100,000
50,000
5
6
7
8
9
10
287
250
230
213
200
172
170
150
128
115
100
85
72
60
48
36
30
24
DIAL SETTING
1 - 2

Section 2

SPECIFICATIONS

2.1 OUTPUTS

OUTPUT	Polarity: Positive Amplitude: 2.5 to 250 mV tail pulses Rise Time: 100 nanoseconds Fall Time Constant: 50 microseconds Impedance: 93 ohms, DC coupled Connector Type: BNC, UG-1094/U, front panel
TRIGGER OUT	Polarity: Positive Amplitude: ≈ 6 volts, (unterminated) logic pulses Rise Time: 100 nanoseconds Fall Time Constant: 50 nanoseconds Pulse Width: ≈ 125 nanoseconds Impedance: 50 ohms, DC coupled Connector Type: BNC UG-1094/U, front panel

2.2 PERFORMANCE

Provides an essentially monoenergetic spectral line (less than 2 keV FWHM peak width) at any given amplitude or count rate

Average output rate continuously variable from less than 5 counts/second to more than 40,000 counts/second guaranteed

Count rate linearity is 2-5% of full scale between 15kHz and 250kHz

No long-term or temperature stability specifications are tested or implied

2.3 CONTROLS

AMPLITUDE	Ten-turn front panel potentiometer used for selecting the output pulse amplitude; range from 25 to 250 mV, (unterminated); both attenuator switches in OUT position
COUNT RATE	Front panel ten-turn potentiometer to select average output rate from less than 5Hz to greater than 240kHz (guaranteed)
ATTENUATOR	Two front panel toggle switches for selecting X2 and X5 attenuation of the normal 250mV full range output, total attenuation X10

2.4 CONNECTORS

TRIGGER OUT	front panel, BNC, UG-1094/U
OUTPUT	front panel, BNC, UG-1094/U

2.5 POWER

+24V	–	40mA	+12V	–	25mA
- 24V	–	25mA	- 12V	–	40mA

2.6 PHYSICAL

SIZE	Standard single-width NIM module (1.35 X 8.714 inches) per TID-20893 (rev.)
NET WEIGHT	2.6 lb (1.17 kg)
SHIPPING WEIGHT	7.6 lb (3.42 kg)

Section 3

CONTROLS AND CONNECTORS

3.1 GENERAL

The functions of the controls and connectors located on the module are described in this section. It is recommended that this section be read before proceeding with the operation of the unit.

3.2 CONTROLS

3.2.1 AMPLITUDE Control (Front Panel)

This control is a ten-turn potentiometer used to vary the output signal over the full output range, providing ease of obtaining an output pulse of the required amplitude. The output can be varied from 25 to 250 mV unterminated.

3.2.2 COUNT RATE Control (Front Panel)

The COUNT RATE control is used to vary the average counting rate from less than 5 counts per second to greater than 240,000 counts per second, utilizing a ten-turn potentiometer. The counting rate is approximately linear over a range from 15,000 counts per second to 250,000 counts per second (See Figure 1-1).

3.2.3 ATTENUATOR Switches (Front Panel)

These toggle switches provide output signal attentuation by factors of 1/2 and 1/5 (or 1/10, when both are used) of the normal 250 mV full-range output.

3.3 CONNECTORS

3.3.1 TRIGGER OUT (Front Panel)

The TRIGGER OUT furnishes positive-going logic pulses to drive a scaler or ratemeter, and provides a timing reference for output pulses.

3.3.2 OUTPUT (Front Panel)

The OUTPUT provides positive-going exponential tail pulses having a rise time of 100 nanoseconds, and a fall time constant of 50 microseconds. The OUTPUT is DC coupled, so that the average DC level of the output signal increases as high count rate pulses begin to pile up.

Section 4

OPERATING INSTRUCTIONS

4.1 GENERAL

The purpose of this section is to familiarize the user with the controls of the Model 1407R Random Noise Pulser, and to check that the unit is operating correctly. Since it is difficult to determine the exact system configuration in which the module will be used, explicit operating instructions cannot be given. However, if the following procedures are carried out, the user will gain sufficient familiarity with this instrument to permit its proper use in the system at hand.

4.2 INITIAL OPERATION

1. Insert module in AEC compatible base unit/power supply such as Canberra Model 1400; turn on power switch.

2. Connect OUTPUT of the Model 1407R to oscilloscope (0.1 V/cm, 100 microsec/cm).

3. Connect TRIGGER output to external trigger of oscilloscope.

4. Switch ATTENUATOR controls to OUT position.

5. Set AMPLITUDE control to maximum (10.0).

6. Set COUNT RATE control to minimum (0.0).

4.3 INITIAL CHECKOUT

1. Rotate the COUNT RATE control slowly clockwise, observing on the oscilloscope the tail pulses which occur at random intervals at an ever-increasing average counting rate. Observe that as the count rate goes above 20,000 counts/second the pulses exhibit a tendency to climb on the tails of each other, and the baseline is lifted from zero to some average positive level, reaching approximately 2.5 VDC as a counting rate of 240,000 counts/second is approached.

2. Turn the COUNT RATE control to 1.00, which should give an approximate average counting rate of 50 counts/second.

3. Turn oscilloscope time base to 1 microsec/cm and observe the amplitude of a single output pulse of 250 mV.

4. Reduce AMPLITUDE control to 0.0; the output amplitude should fall to about 25 mV.

5. Return AMPLITUDE control to maximum, and operate the ATTENUATOR switches, observing the decrease in the output signal proportional to the attenuation selected.

4.4 TYPICAL APPLICATIONS

The Model 1407R can be used to investigate the effect of a high counting rate on linear or biased amplifiers, DC restorers, single channel analyzers, etc. By connecting the OUTPUT of the Model 1407R to the positive input of, for example, a linear amplifier, degradation in pulse shape, baseline shift, and pile up can be observed for different average counting rates. Similarly, these effects could be shown with a biased amplifier, DC restorer, or other linear signal processing modules.

By connecting the TRIGGER output of the Model 1407R to a scaler, the actual number of counts generated by the unit will be obtained, and comparison may be made with an observed count rate from a linear amplifier at different pulse-shaping time constants. In this way the effect of pulse pile-up on the average count rate can be seen.

Similarly, when using a linear amplifier and a single channel analyzer, the difference in observed counts between the Model 1407R, the linear amplifier, and the single channel analyzer will show the relative effects of pulse pile-up in the amplifier, and the combination of dead time and pulse pair resolution in the single channel analyzer.

Section 5

CIRCUIT DESCRIPTION

The pulse rate of the Model 1407R originates as random thermal noise generated by a single resistor. The amplitude of the noise signals thus generated is then compared to the setting obtained from a variable bias, the COUNT RATE control. As the setting is decreased, the number of noise spikes exceeding the setting becomes greater, increasing the count rate.

The output of the comparator triggers a 100 nanosecond monostable, which in turn generates the TRIGGER output. The output of the monostable also opens a gate, allowing a variable constant current source (the AMPLITUDE control) to charge a storage capacitor.

An output amplifier provides pulse shaping and line-driving capability. Attenuation circuitry permits full-scale output to be reduced by factors of 1/2, 1/5, or 1/10 from the 250 mV nominal.

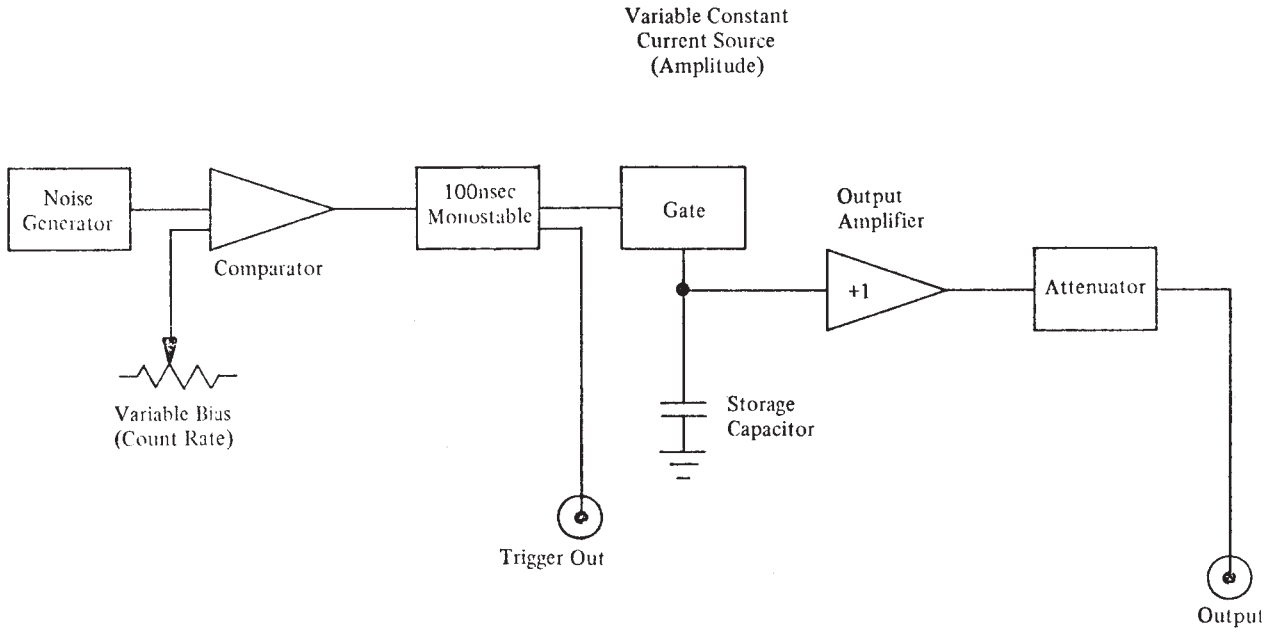
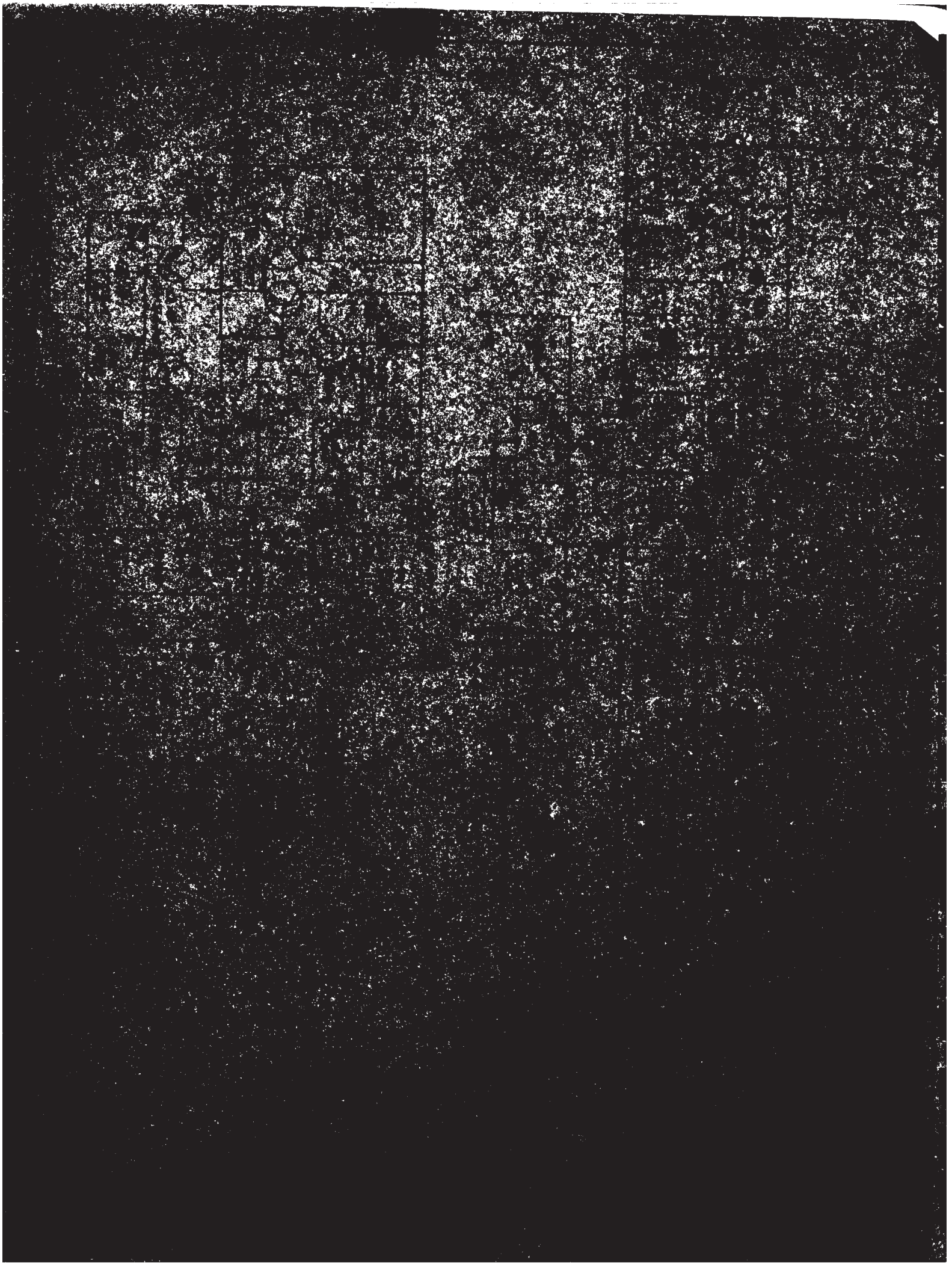


Figure 5-1. Block Diagram, Model 1407R



MODEL 207X-03

EIA INTERFACE

OPERATORS MANUAL

V 2.6

TABLE OF CONTENTS

SECTION		PAGE
1	INTRODUCTION	1
1.1	OPTION 207X-03 FEATURES	1
1.2	DESCRIPTION	1
1.3	CONTROLS	1
1.4	CONTROL COMMANDS	2
2	SPECIFICATIONS	2
2.1	OUTPUT	2
2.2	REAR PANEL OUTPUT CONNECTOR	2
2.3	CABLES	3
2.4	POWER	3
3	OPERATION	4
3.1	FIELD INSTALLATION OF OPTION BOARD	4
3.2	EIA CHAIN CABLE INSTALLATION	4
4	SINGLE UNIT OPERATING MODES	5
4.1	SIMPLE PRINTER/COMPUTER READOUT	5
4.2	COMPUTER CONTROL	5
5	CHAIN OPERATING MODES	7
5.1	MASTER/SLAVE OPERATION	7
5.2	SYNCHRONOUS MASTER/SLAVE OPERATION	8
5.3	MULTIPLEXED SUBGROUPS	9
5.4	DIRECT COMPUTER CONTROL	10
5.5	SIMPLE READOUT	11
5.6	SIMPLE READOUT-MULTIPLE SUBGROUPS	12
6	CONFIGURATION DIAGRAMS	13
7	READOUT FORMATS	15
7.1	SINGLE UNIT READOUT FORMAT	15
7.2	READOUT FORMAT CHAIN MODE	17
APPENDIX A - INTERNAL CONTROLS		20
APPENDIX B - ADVANCED FEATURES EIA V2.6		22
APPENDIX C - CABLE WIRING AND SIGNALS		24
APPENDIX D - COMPONENT LAYOUT		26
APPENDIX E - SCHEMATIC		27

1 INTRODUCTION

The Canberra EIA Interface -03 option provides for readout to a EIA compatible printer and computer control of start, stop reset readout via a single cable.

1.1 OPTION 207X-03 FEATURES

- Provides 8 digit readout to RS232C EIA device
- Assigns I.D. numbers to chain of 1-99 units
- Field installable
- Multiplexes independent counting systems
- Provides for computer control of Start, Stop, Reset and Readout

1.2 DESCRIPTION

The EIA Interface option provides the 207X module (2071, 2072 2071A or 2072A) with RS323C EIA capabilities for readout to print terminal or computer. It also provides for external control from terminal or computer. Multiple 207X series modules, with -03 opti in each one, can be chained together with one EIA line. This all the terminal or computer to control or readout modules on the l either individually or all at once, making multiple Master/Sl operation or multiplexing independent experiments on the single line possible.

1.3 CONTROLS

Side panel cutout provides access to:

- EIA single/recycle operation
- enable/disable formfeeds, linefeeds and carriage returns readout format
- select Baud rate
- select EIA line chain operation
- select unit as Master or slave

1.4 CONTROL COMMANDS

Control Commands from a terminal or computer include:
- Start, Stop, and Reset of the module.
- Readout of module when stopped, during counting cycle, or sto
  to buffer internal to the -03 option for future readout.
- Change readout format leading zero suppresssion, CR after
  number, variable pause after each character and/or CR, and del
  response to readout command.
- Address all or one individual module on EIA chain.
- Select XON/XOFF software handshake.

2 SPECIFICATIONS

2.1 OUTPUT

- Baud rate switch selectable to 150, 300, 600, 1200, 2400, 4
  9600, 19200.
- Fixed 8 bit word length, no parity, 2 stop bits.
- Hardware or software handshake.
- EIA chain of 1 to 99 modules.

2.2 REAR PANEL OUTPUT CONNECTOR

-Rear panel output is 25 pin "D" type female connector wired a
follows:

PIN	EIA	DESCRIPTION	COMMENT (CONNECTED AS DATASET)
1	AA	-	Protective Ground
2	BA	TXD	Data input
3	BB	RXD	Data output
4	CA	RTS	Must be high (> +5V)
5	CB	CTS	High when ready to receive
6	CC	DSR	High when ready to receive
7	AB	GND	Signal ground
8	CF	DCD	Space when unit is switched on
11	Flag	-	Used in chain mode (last unit)
20	CD	DTR	Must be high (> +5V)
14	-	-	+12V pull up

If 4 or 20 is not used connect to pin 14.

2.3 CABLES

- Cables additional cost option.
- Specify C1535 for single module cable or C1536 for one to f modules on an EIA chain.
- For cables connecting more than four units contact the factory.

2.4 POWER

+12V--Additional 70 mA
-12V--Additional 70 mA
+24V--Not used
-24V--Not used

3 OPERATION

The purpose of this section is to familiarize the user with th operation and use of the Model 207X-03 EIA Interface.

3.1 FIELD INSTALLATION OF OPTION BOARD

--The EIA Interface option consists of: two standoffs, tw lockwashers, four Phillips-head screws, and the interface board.

To install the option, first remove both side covers, each of which held in place by five Phillips-head screws. Then:

1. If they are not already installed, install the two standoffs the rearmost two holes in the unit's printed circuit board.

2. The standoffs are to be placed on the component side of the bvc and fastended to the board with one lockwasher and one screw ea

3. Carefully place the interface board in the unit so that the twe five pin EIA connector is at the rear of the board.

4. With the interface board's two mounting holes lined up with two standoffs, press the board gently down to mate the interbo connnectors.

   The interface board should mate smoothly with the uni connector. If any resistance is felt, the connector is properly aligned; lift the interface board up a little and reali the two mounting holes with the standofffs before pressing d again.

5. Fasten the interface board to the standoffs with two screws, in each corner of the interface board.

6. Before replacing the side covers, check the intallation for pro operation.

3.2 EIA CHAIN CABLE INSTALLATION

1. The C1536 EIA chain cable option provides one cable allowing user to connect from one to four units. Three terminator plu are supplied with this cable.

2. If the C1536 cable is to be used with only one unit, then t
unit must be connected to the UNIT 1 connector on the cab
Terminator plugs must be connected to the three unused connecto
Note that the EIA chain switch SW10 on the Option board must
OFF for this mode of operation.

3. If the C1536 cable is to be used with more than one unit, but l
than four units then one of the units must be connected to UNI
(the first unit in the chain) and one unit must be connected
UNIT 4 (the last unit in the chain). Terminator plugs MUST
connected to all unused connectors in between. Note for this m
of operation chain switch SW10 on the Option board must be ON
all units.

See Appendix C for cable description and connection to printer
computer.

4 SINGLE UNIT OPERATING MODES

4.1 SIMPLE PRINTER/COMPUTER READOUT

--used to send data from the unit to a computer or printer withou
direct control from a computer. The unit is started from it's fr
panel, and automatically sends data to the printer/computer when
reaches preset or is stopped from the front panel.

With EIA recycle selected on the unit's option -03 board (S1-ON)
unit will readout, clear data, and restart upon the reaching fr
panel preset. With front panel RECYCLE selected, stopping of the u
by any method will cause a readout/clear data/restart. The Fr
Panel Recycle and the Option 03 Recycle switches should not both
set on.

- Switch settings for this operation are -

S1 (Recycle) OFF
S2-S8 (Readout format and Baud rate) as required
S9 (Master enable) OFF
S10 (Chain Mode) OFF

4.2 COMPUTER CONTROL

--the unit is Started, Stopped, Reset or Readout by a simp
command from the computer. If a front panel preset is selected,
the unit will automatically read out when the preset is reached.

With EIA recycle selected on the unit's option -03 board (S1-ON) th
unit will readout, clear data, and restart upon the reaching fron
panel preset. With front panel RECYCLE selected, stopping of the uni
by any method will cause a readout/clear data/restart. The Fron
Panel Recycle and the Option 03 Recycle switches should not both
set on.

- Switch settings for this operation are -

S1 (Recycle) OFF
S2-S8 (Readout format and Baud rate) as required
S9 (Master enable) OFF
S10 (Chain Mode) OFF

* Basic Computer Control Commands *

COMMAND	CODE	HEX	DECIMAL	ACTION
START	CONTROL-X	18	24	Clear unit, start counting
STOP	CONTROL-Y	19	25	Stop unit
READOUT	CONTROL-W	17	23	Readout unit (counting or stopped)

5 CHAIN OPERATING MODES

5.1 MASTER/SLAVE OPERATION

--the computer controls the Master (which must be Unit #1 with S set ON) in the EIA chain and the Master then controls the remain units (the Slaves) via internal commands on the EIA chain.

All units on the EIA chain are Started, Stopped, Reset or Readout b simple command from the computer. All units on the EIA chain st and stop at the same time with the exception of a small unit to u communications delay time. The delay is determined by the selec baud rate - for example, with 4 units in the chain the delay betw starting the first unit and starting the last unit is 2 milliseco at 19.2k Baud.

If a front panel preset is selected on the Master unit, all units the chain will stop, and automatically read out, when that preset reached. If a preset is selected on a Slave unit, only the Slave w stop and automatically read out when the preset is reached. Manua stopping a unit on the front panel has the same effect as reach preset.

With EIA recycle selected on the Master option -03 board (S1-ON) units on the chain will readout, clear data, and restart upon Master reaching front panel preset. A computer stop command will cause a readout/clear data/restart. With front panel RECYCLE selec on the Master, stopping the Master by any method will cause readout/clear data/restart. The Front Panel Recycle and the Option Recycle recycle switches should not both be set on.

- Switch settings for this operation are -

S1 (Recycle) ON or OFF in Master as required, OFF on all Slaves
S2-S8 (Readout format and Baud rate) as required - all units set t same
S9 (Master enable) ON in Master (Unit #1) and OFF in all Slaves
S10 (Chain Mode) ON in all units

* Basic Computer Control Commands *

COMMAND	CODE	HEX	DECIMAL	ACTION
START	CONTROL-X	18	24	Clear all units, start countin
STOP	CONTROL-Y	19	25	Stop all units
READOUT	CONTROL-W	17	23	Readout all units (counting or stopped)

5.2 SYNCHRONOUS MASTER/SLAVE OPERATION

--all units in the chain have their Front Panel or Rear Pane ENABLE lines connected together using coax cables , and Unit #1 been set as the Master by the computer transmitting a CONTROL-D to EIA chain. Master enable switch S9 must be off in all units. computer then controls the Master via the EIA chain, and the Mas controls the Slaves via the ENABLE line providing simultaneous St and Stopping.

All units on the EIA chain are Started, Stopped, Reset or Readout b simple command from the computer. All units are Started and Stop synchronously. Readout also occurs automatically when the EIA ch is Stopped.

If a front panel preset is selected on the Master unit, all units the chain will stop, and automatically read out, when that preset reached. Slave units cannot have a preset selected and should h N=M=10^9=0 on their front panel. Manually stopping the Master on the front panel has the same effect as reaching pres Stopping the Slaves from the front panel has no effect on the chain

With EIA recycle selected on the Master option -03 board (S1-ON) units on the chain will readout, clear data, and restart upon Master reaching front panel preset. Computer stop command will s the units, read them all out, but will not start them ag automatically. With front panel RECYCLE selected on the Mast stopping the Master by any method will cause a readout/cl data/restart. The Front Panel Recycle and the Option 03 Recy switches should not both be set on.

- Switch settings for this operation are -

S1 (Recycle) OFF in Slaves - ON or OFF in Master as required
S2-S8 (Readout format and Baud rate) as required-all units set the same
S9 (Master enable) OFF in all units
S10 (Chain Mode) ON in all units

* Basic Computer Control Commands *

COMMAND	CODE	HEX	DECIMAL	ACTION
CCM	CONTROL-D	4	4	Selects ENABLE MASTER/SLAVE operation. Must be sent first upon power up of units to ena this type operation.
START	CONTROL-X	18	24	Clear all units, start counti
STOP	CONTROL-Y	19	25	Stop all units
READOUT	CONTROL-W	17	23	Readout all units (counting o stopped)

5.3 MULTIPLEXED SUBGROUPS

--operation possible by connecting subgroups of units together wit
their ENABLE lines. A subgroup is connected to another subgroup o
by the EIA chain. Control of each subgroup is accomplis
independently by the computer addressing a single unit in t
subgroup. This unit becomes Master in that subgroup, and controls
Slaves via the ENABLE line.

All units in the addressed subgroup are Started, Stopped , Reset
Readout by a simple command from the computer to the addressed Mast
All subgroup units are Started and Stopped synchronously. Readout
all subgroup units BUT THE MASTER occurs automatically when
subgroup Master is stopped by the computer. The Master alone can
readout by addressing it and reading it out.

If a front panel preset is selected on the Master unit, all units
that subgroup will stop, and automatically read out, when that pre
is reached. Slave units CANNOT have a preset selected and should h
$N=M=10^{P}=0$ on their front panel. Manually stopping the
Master on the front panel has the same effect as reaching preset.

With EIA recycle selected on the Master option -03 board (S1-ON) un
in a subgroup will readout, clear data, and restart upon the Mas
reaching front panel preset. Computer stop command will stop
units, read out all units BUT THE MASTER, but will not start t
again automatically. With front panel RECYCLE selected, stopping
the unit by any method will cause a readout/clear data/restart.
Front Panel Recycle and the Option 03 Recycle switches should not b
be set on.

- Switch settings for this operation are -

S1 (Recycle) OFF in Slaves - ON or OFF on Master as required
S2-S8 (Readout format and Baud rate) as required - all units set t
same
S9 (Master enable) OFF in all units
S10 (Chain Mode) ON in all units

* Basic Computer Control Commands *

COMMAND	CODE	HEX	DECIMAL	ACTION
AM ON	ESC nn	1B,n,n	27,n,n	Address unit #nn as Subgro Master (1)
START	CONTROL-X	18	24	Clear all units in Subgrou and start them counting
STOP	CONTROL-Y	19	25	Stop all units in Subgroup
READOUT	CONTROL-W	17	23	Readout Master unit (count or stopped)
AM OFF	ESC 00	1B,30,30	27,48,48	Address all units (2)

(1) For Example, Unit 5 would be addressed with an ESC 05 (Hex 1B, 3
35 or Decimal 27, 48, 53) -- leading zero in unit number is required

(2) READOUT (CONTROL-W) will read out all Subgroups at once in th
mode. Starting and stopping of Chain with START (CONTROL-X) or ST
(CONTROL-Y) is not valid operation in this mode.

5.4 DIRECT COMPUTER CONTROL

--used where Master/Slave operation is not required. Either the
complete chain or individual units on the chain can be Starte
Stopped, Reset or Readout by commands from the computer.

Once started by the computer, individual units will automatically re
out once they reach preset (if any). If the EIA or front pan
Recycle and a preset are selected on an individual unit, that un
will, upon reaching preset, Stop, Readout, Clear Data, and Sta
Again. Manually stopping a unit on the front panel has the sa
effect as reaching preset. With front panel RECYCLE selecte
stopping of the unit by any method will cause a readout/cle
data/restart. The Front Panel Recycle and the Option 03 Recyc
switches should not both be set on.

- Switch settings for this operation are -

S1 (Recycle) ON or OFF in any unit as required
S2-S8 (Readout format and Baud rate) as required - all units set th
same
S9 (Master enable) OFF in all units
S10 (Chain Mode) ON in all units

* Basic Computer Control Commands *

COMMAND	CODE	HEX	DECIMAL	ACTION
AM ON	ESC nn	1B,n,n	27,n,n	Address unit #nn in Chain
AM OFF	ESC 00	1B,30,30	27,48,48	Address all units
START	CONTROL-X	18	24	Clear addressed unit(s) in chain and start unit(s) counting. (2)
STOP	CONTROL-Y	19	25	Stop addressed unit(s). (2
READOUT	CONTROL-W	17	23	Readout addressed unit(s) (counting or stopped). (2

(1) For Example, Unit 5 would be addressed with an ESC 05 (Hex 1B,
35 or Decimal 27, 48, 53) -- leading zero in unit number is require

(2) Either all units or only one are addressed.

5.5 SIMPLE READOUT

--to send data from the EIA chain to a computer or printer witho
direct control from a computer. All units on the chain are sta
from their front panel, and automatically send data to
printer/computer when they reach preset or are stopped from the f
panel. Simultaneous starting and stopping of all units can
accomplished by connecting the units together via the ENABLE line.

With EIA recycle selected on the individual unit's option -03 bo
(S1-ON) the unit will readout, clear data, and restart upon
reaching front panel preset. With front panel RECYCLE select
stopping of the unit by any method will cause a readout/cl
data/restart. The Front Panel Recycle and the Option 03 Recy
switches should not both be set on.

- Switch settings for this operation are -

S1 (Recycle) ON or OFF in all units as required
S2-S8 (Readout format, Baud rate) as required; all units set the s
S9 (Master enable) OFF in all units
S10 (Chain Mode) ON in all units

5.6 SIMPLE READOUT-MULTIPLE SUBGROUPS

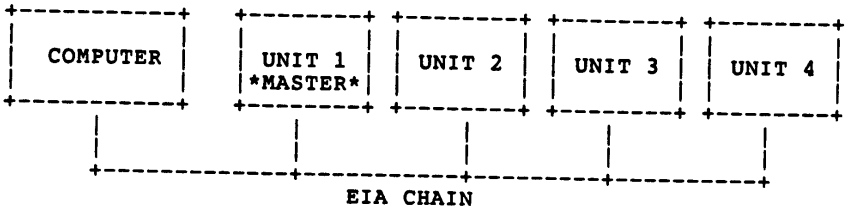
--to operate and send data from independent subgroups to a computer or printer without direct control from a computer. Operation possib by connecting subgroups of units together with their ENABLE lines. subgroup is connected to another subgroup only by the EIA chai Control of each subgroup is accomplished independently by starting unit in a subgroup with its front panel Start. This unit becom Master in that subgroup, and controls the Slaves via the ENABLE lin When a unit in a subgroup is started this way, all units in th subgroup start and automatically send data to the printer/comput when the Master reaches preset or is stopped from the front panel.

With EIA recycle selected on the individual Master unit's option - board (S1-ON) the subgroup will readout, clear data, and restart up the Master reaching front panel preset. With front panel RECYC selected, stopping of the Master by any method will cause readout/clear data/restart. The Front Panel Recycle and the Option Recycle switches should not both be set on.

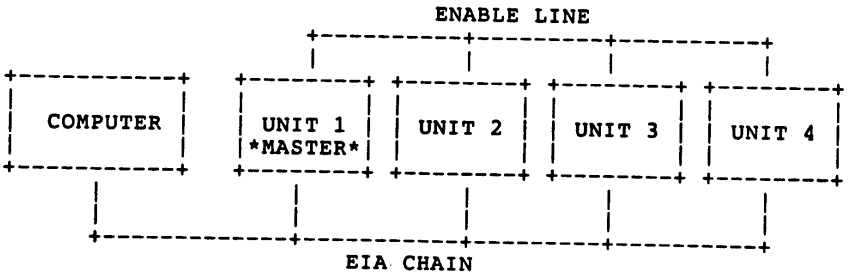
- Switch settings for this operation are -

S1 (Recycle) ON or OFF in only the Master unit as required
OFF on all Slaves
S2-S8 (Readout format, Baud rate) as required; all units set the sa
S9 (Master enable) OFF in all units
S10 (Chain Mode) ON in all units

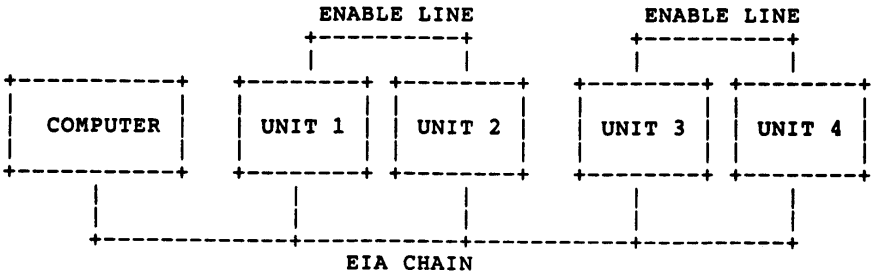
6 CONFIGURATION DIAGRAMS



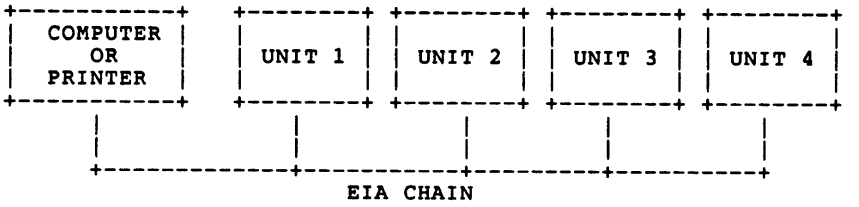
MASTER/SLAVE OPERATION



SYNCHRONOUS MASTER/SLAVE OPERATION



MULTIPLEXED SUBGROUPS
or
SIMPLE READOUT-MULTIPLE SUBGROUPS



DIRECT COMPUTER CONTROL
or
SIMPLE READOUT

7 READOUT FORMATS

7.1 SINGLE UNIT READOUT FORMAT

- When first powered up the unit transmits:

```
*
*
SP
E
I
A
SP
2       :
.       :2.6 OR LATEST FIRMWARE VERSION
6       :
SP
*
*
CR
LF
FF
```

- All normal readouts in the Single Unit mode use X-ON/X- handshake. The unit transmits an X-ON (Control Q, Hex 11) start of data transmission, and transmits an X-OFF (Control S, 13) at the end of transmission.

- Normal readout when unit is stopped by Front Panel STOP, hav preset reached, or by command from the computer:

```
D8A     : MSD OF CHANNEL A
D7A
D6A
D5A
D4A
D3A
D2A
D1A     : LSD OF CHANNEL A
SP      : IF SW5 ON
SP      : IF SW5 ON
D8B     : MSD OF CHANNEL B
D7B
D6B
D5B
D4B
```

```
D3B
D2B
D1B     : LSD OF CHANNEL B
CR      : IF SW4 ON
PSE     : PAUSE IF ENABLED
LF      : IF SW3 ON
FF      : IF SW2 ON
```

o If CR-DELIM is selected with a CONTROL-K sent from the comp
after power up, then there is added Carriage Returns as follow

```
D8A     : MSD OF CHANNEL A
D7A
D6A
D5A
D4A
D3A
D2A
D1A     : LSD OF CHANNEL A
CR      : IF SW4 ON
LF      : IF SW3 ON
FF      : IF SW2 ON
D8B     : MSD OF CHANNEL B
D7B
D6B
D5B
D4B
D3B
D2B
D1B     : LSD OF CHANNEL B
CR      : IF SW4 ON
LF      : IF SW3 ON
FF      : IF SW2 ON
```

7.2 READOUT FORMAT CHAIN MODE

- When first powered up the chain transmits:

```
*
CR
LF
#        : MSD number units in chain (SP if <10 units)
#        : LSD number units in chain
SP
U
N
I
T
S
SP
I
N
SP
C
H
A
I
N
CR
LF
```

- NORMAL OUTPUT SEQUENCE

1. Without the use of the ENABLE signal, readout of the units always in an ascending order (ie: Unit 1, Unit 2, ... U N).

2. With the use of the ENABLE signal, a readout using CONTROL-W command will be in ascending order but all ot modes of readout will result in the units reading out decending order (Unit N ... Unit 2, Unit 1).

3. In Multiple Subgroup modes where the Subgroups independently stopped by reaching front panel presets, i second Subgroup reaches preset as another is in the process reading out, it is possible for the second Subgroup readout interpose itself on the first. For example, if Subgroup A Units 1 and 2 and Subgroup B is Units 3 and 4, it is possi to have a readout sequence of 2, 4, 3, 1. Here Subgroup reached preset while Subgroup A was reading out, and

normal 4, 3 order of Subgroup B is interposed inside the 2, order of Subgroup A readout.

4. Normal readout format for each unit is as follows:

```
CH#      : MSD OF UNIT NUMBER
CH#      : LSD OF UNIT NUMBER
:
SP       : IF SW5 ON
SP       : IF SW5 ON
D8A      : MSD OF CHANNEL A
D7A
D6A
D5A
D4A
D3A
D2A
D1A      : LSD OF CHANNEL A
SP       : IF SW5 ON
SP       : IF SW5 ON
D8B      : MSD OF CHANNEL B
D7B
D6B
D5B
D4B
D3B
D2B
D1B      : LSD OF CHANNEL B
CR       : IF SW4 ON
PSE      : PAUSE IF ENABLED
LF       : IF SW3 ON
FF       : IF SW2 ON
```

5. If CR-DELIM is selected with a CONTROL-K sent to units af
power up, then there is added Carriage Returns as follows:

```
CH#     : MSD OF UNIT NUMBER
CH#     : LSD OF UNIT NUMBER
CR      : IF SW4 ON
PSE     : PAUSE IF ENABLED
LF      : IF SW3 ON
FF      : IF SW2 ON
D8A     : MSD OF CHANNEL A
D7A
D6A
D5A
D4A
D3A
D2A
D1A     : LSD OF CHANNEL A
CR      : IF SW4 ON
PSE     : PAUSE IF ENABLED
LF      : IF SW3 ON
FF      : IF SW2 ON
D8B     : MSD OF CHANNEL B
D7B
D6B
D5B
D4B
D3B
D2B
D1B     : LSD OF CHANNEL B
CR      : IF SW4 ON
PSE     : PAUSE IF ENABLED
LF      : IF SW3 ON
FF      : IF SW2 ON
```

APPENDIX A

A. INTERNAL CONTROLS

The ten-section DIP switch on the option board is accessible throu
the hole in the unit's left side cover. Switch functions are
follows:

SW1	RECYCLE/SINGLE CYCLE	OFF
SW2	FF	OFF
SW3	LF	OFF
SW4	CR	OFF
SW5	SP	
SW6	BAUD 2 :	ON
SW7	BAUD 1 : SEE CHART BELOW	ON
SW8	BAUD 0 :	OFF
SW9	ECHO/MASTER ENABLE	OFF
SW10	EIA CHAIN/SINGLE UNIT	OFF

B. CONTROLS DESCRIPTION

* SW1 OFF-Selects Single Cycle mode. When unit stops counting it will readout, but not start up again unless the Front Panel SINGLE/RECYCLE is in the RECYCLE position.
  SW2 ON-Selects Recycle mode. When unit stops counting, it will readout, clear, and start counting. Front Panel RECYCLE cannot be selected while in this mode.

* SW2 OFF-No Formfeeds generated during readout (see Section 7).
  SW2 ON-Formfeeds generated during readout.

  SW3 OFF-No Linefeeds generated during readout (see Section 7).
* SW3 ON-linefeeds generated during readout.

  SW4 OFF-No Carriage Returns generated during readout (see Section 7).
* SW4 ON-Carriage Returns generated during readout.

  SW5 OFF-No Spaces generated during readout (see Section 7).
* SW5 ON-Spaces generated during readout.

SW6, SW7, SW8 BAUD RATE SETTINGS

BAUD	SW6	SW7	SW8
19200	ON	ON	ON
9600	ON	ON	OFF
4800	ON	OFF	ON
2400	ON	OFF	OFF
1200	OFF	ON	ON
600	OFF	ON	OFF
* 300	OFF	OFF	ON
150	OFF	OFF	OFF

* SW9 OFF-In EIA Chain mode (SW10-ON) unit not a master. In Single Mode characters sent to the -03 are not echoed back.

SW9 ON- In EIA Chain mode, unit is master (select SW9 ON in Unit # 1 only). In Single Mode characters sent to the -03 from the computer are echoed back to the computer.

* SW10 OFF-Selects Single Unit mode of operation, no chain operation allowed.

SW10 ON-Selects EIA Chain mode of operation, all units in chain must have this switch ON.

* Indicates Factory default setting of switches.

B. ADVANCED FEATURES EIA INTERFACE VERSION 2.6

XOFF/XON PROTOCOL: ^S / ^Q (IF SW10 OFF OR WHEN CHAIN-^S/^Q ENABLED)

CONTROLCOMMANDS:

<	START: ^X	
C/T <<	STOP: ^Y	(+READIN)
<	RESET: ^Z	
	READIN: ^R	(PUT DATA FROM COUNTER TO EIA-OPTION) (WITHOUT STOPPING THE COUNTER)
	READOUT: ^W	(PUT DATA FROM EIA-OPTION TO OUTPUTDEVICE) (WITHOUT STOPPING THE COUNTER)
	LZS OFF: ^U	(LEADING ZERO SUPPRESSION OFF)
	LZS ON: ^V	(DEFAULT)
	CR DELIM.^K	SEND CR AFTER EACH NUMBER (NOTE 3)
	CCMB ^D	COMPUTER CONTROL OF MASTER IN MODE B
	PSE OFF: ^\	(NO PAUSE AFTER CR) (==ESC-M-0-0)
	PSE ON: ^F	PAUSE AFTER CR (32 msec) (DEFAULT) (==ESC-M-2-0)
	^W OFF: ^I	DISABLE AUTO READOUT (^W) IN MODE A
	^W ON: ^E	ENABLE AUTO READOUT (^W) IN MODE A (DEFAULT)
	CHAIN-^S/^Q ON: ^A	ENABLE ^S/^Q-PROTOCOL IN CHAIN MODE (NOTE 1)
	CHAIN-^S/^Q OFF: ^B	DISABLE ^S/^Q-PROTOCOL IN CHAIN MODE (DEFAULT)
	LLO: ^]	SAME AS ESC^X (SEE NOTE 2)
	NLLO: ^^	SAME AS ESC^Y (SEE NOTE 2)
	ESC^D%%	PAUSE AFTER EACH CHARACTER (0 TO 255 msec) (DEFAULT: 0 msec)
	ESC M%%	PAUSE AFTER CARRIAGE RET. (0 TO 255 msec) (DEFAULT: 32 msec) (hex:20)
	ESC W%%	DELAYED RESPONSE TO ^W (0 TO 255 msec) (DEFAULT: 0 msec)
DDO:	ESC^S	DISABLE DATA OUTPUT
EDO:	ESC^Q	ENABLE DATA OUTPUT (DEFAULT)
RESET:	ESC^R	MASTER RESET
LLO:	ESC^X	(LOCAL LOCK OUT) DISABLE LOCAL INTERRUPT
NLLO:	ESC^Y	(NO LOCAL LOCK OUT) ENABLE LOCAL INTERRUPT
RINIF:	ESC^Z	RESET INI & MASTER FLAG
AM ON:	ESC##	ADDRESS MODE ON (ACTIVATE UNIT ##) (NOTE 2)
AM OFF:	ESC00	ADDRESS MODE OFF (ACTIVATE ALL UNITS)
	ESC??	DISABLE ALL UNITS

UNIT NUMBER (01 TO 99)
%% 2-digit HEX NUMBER (00 TO FF)

NOTE 1: ^S/^Q-PROTOCOL CAN BE USE IN CHAIN MODE IF ENABLED BY ^A-COMMAND. WHEN THE MASTER-UNIT RECEIVES A ^S-CHARACTER, IT PUTS A LOW LEVEL TO EIA-LINE 8 (DCD) AND PUTS A HIGH LEVEL TO LINE 8 AFTER A ^Q-CHARACTER. IF A CONNECTION IS MADE FROM MASTER EIA PIN 8 TO LAST SLAVE'S EIA PIN 20 (DTR) THE DATAFLOW CAN ALSO IN THE CHAIN MODE BE CONTROLLED BY ^S/^Q

NOTE 2: IN THE ADDRESS MODE ONLY THE SELECTED UNIT RESPONDS TO CONTROL COMMANDS. STILL ALL UNITS RESPOND TO ESC-SEQUENCES

NOTE 3: WHEN CR-DELIMITER IS SELECTED THE OUTPUTSEQUENCE IS: CHANNELNUMBER,CR-LF-FF,CHANNEL A,CR-LF-FF,CANNEL B, CR-LF-FF (AS SELECTED BY DIPSWITCHES). BLANKS AND COLON ARE DISABLED

APPENDIX B (Continued)

MODE:

	SW1:	
SINGLE	OFF	
RECYCLE	ON	(AUTO-START AFTER STOP BY PB OR TIME PRESET

	SW10:	
SINGLE	OFF	
CHAIN	ON	OPERATION OF MULTIPLE C/T UNITS VIA A SINGLE EIA-LINE

NOTE: WHEN RECYCLE ON THE OPT. 207X-03 IS ENABLED DEPRESSING THE STOP-PB WILL NOT STOP THE C/T BUT INITIATE A READOUT (& RESET & START).
TO STOP THE COUNTER HOLD "RESET"-PB DEPRESSED AND THEN PRES THE "STOP"-PB !!
(THIS IS WHY ANY STOP ACTION WILL PUT THE C/T BACK INTO THE COUNTING MODE BY THE OPT. 207X-03 WHEN RECYCLE IS ENABLED)

ECHO:

SW9:	(NOT IN CHAIN MODE)
ON	ON
OFF	OFF

MASTER ENABLE:

SW9:	(ONLY IN CHAIN MODE)
ENABLED ON	MASTER C/T CAN CONTROL SLAVE UNITS V EIA INTERFACE (i.e. CAN OUTPUT ^X,^V (^W ONLY OF NOT SUPPRESSED BY ^I-COM
DISABLED OFF	

BAUDRATE:

	SW6	SW7	SW8
19200 BAUD:	ON	ON	ON
9600 BAUD:	ON	ON	OFF
4800 BAUD:	ON	OFF	ON
2400 BAUD:	ON	OFF	OFF
1200 BAUD:	OFF	ON	ON
600 BAUD:	OFF	ON	OFF
300 BAUD:	OFF	OFF	ON
150 BAUD:	OFF	OFF	OFF

WORDLENGTH: 8 BIT

APPENDIX C

C. CABLE WIRING AND SIGNALS

1. Rear panel output is 25 pin "D" type female connector wired follows:

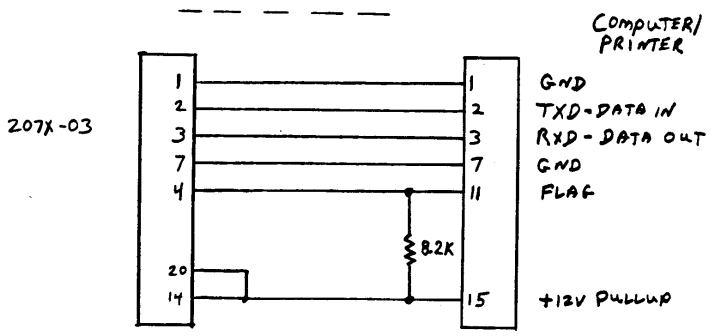
PIN	EIA	DESCRIPTION	COMMENT (CONNECTED AS DATASET)
1	AA	-	Protective Ground
2	BA	TXD	Data input
3	BB	RXD	Data output
4	CA	RTS	Must be high (> +5V)
5	CB	CTS	High when ready to receive
6	CC	DSR	High when ready to receive
7	AB	GND	Signal ground
8	CF	DCD	Space when unit is switched on
11	Flag	-	Used in chain mode (last unit)
20	CD	DTR	Must be high (> +5V)
14	-	-	+12V pull up

Pins 4 or 20 can be used for hardware handshake. If the pin is connect to pin 14.

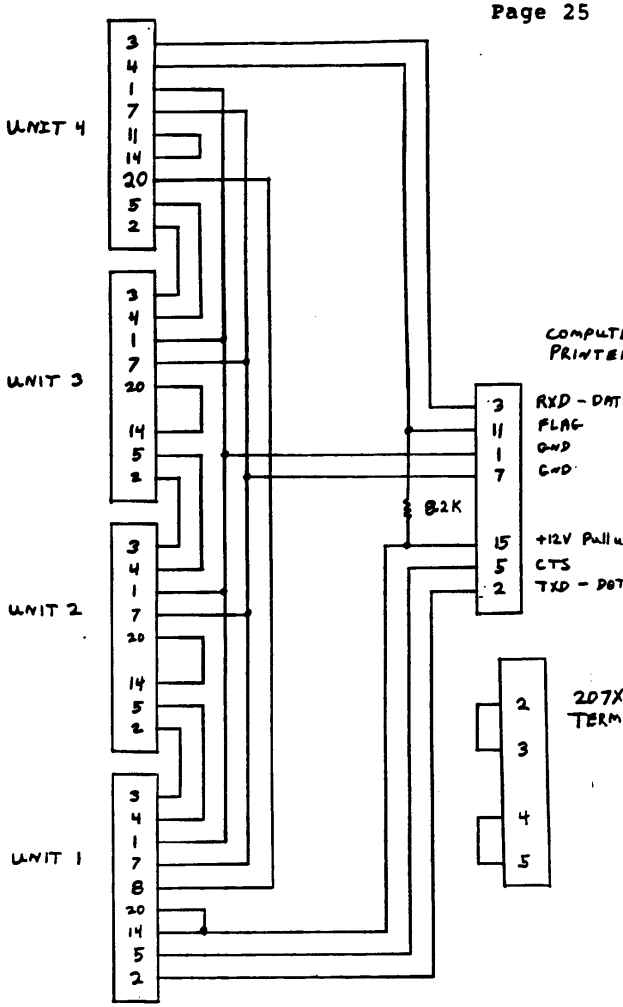
2. The COMPUTER/PRINTER end of the optional C1535 Single Module C1536 EIA chain cables is wired to emulate the EIA output port a Canberra MCA, and can be used in conjunction with the follow Canberra cables to interconnect to various devices:

C1500	Decwriter, IBM computer
C1519	IBM Adapter (9 pin)-also requires C1500 cable
C1506	TTY 43
C1507	Data Products Printer
C1509	DEC computer
C1508	Apple II series with Super Serial Card

3. C1535 Cable Wiring:



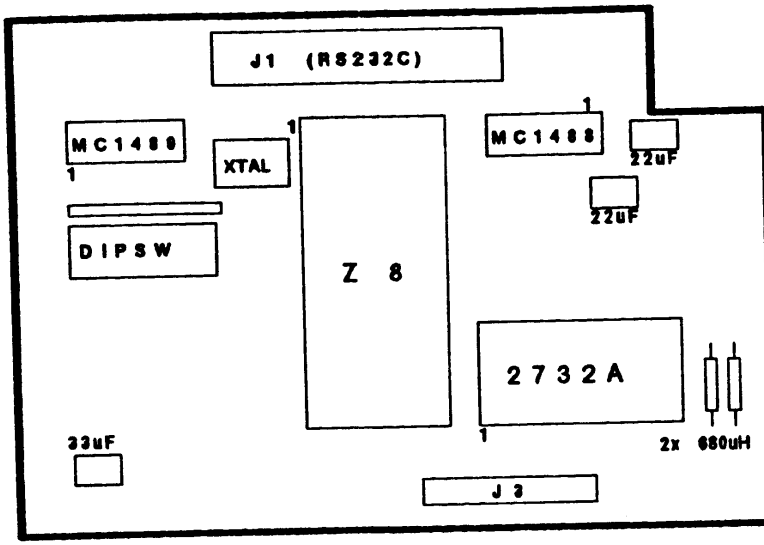
4. C1536 Cable Wiring:



Note that Pin 5 on the COMPUTER/PRINTER connector is used Hardware handshake, and must be left open if X-ON/X-OFF handsh protocol is used.

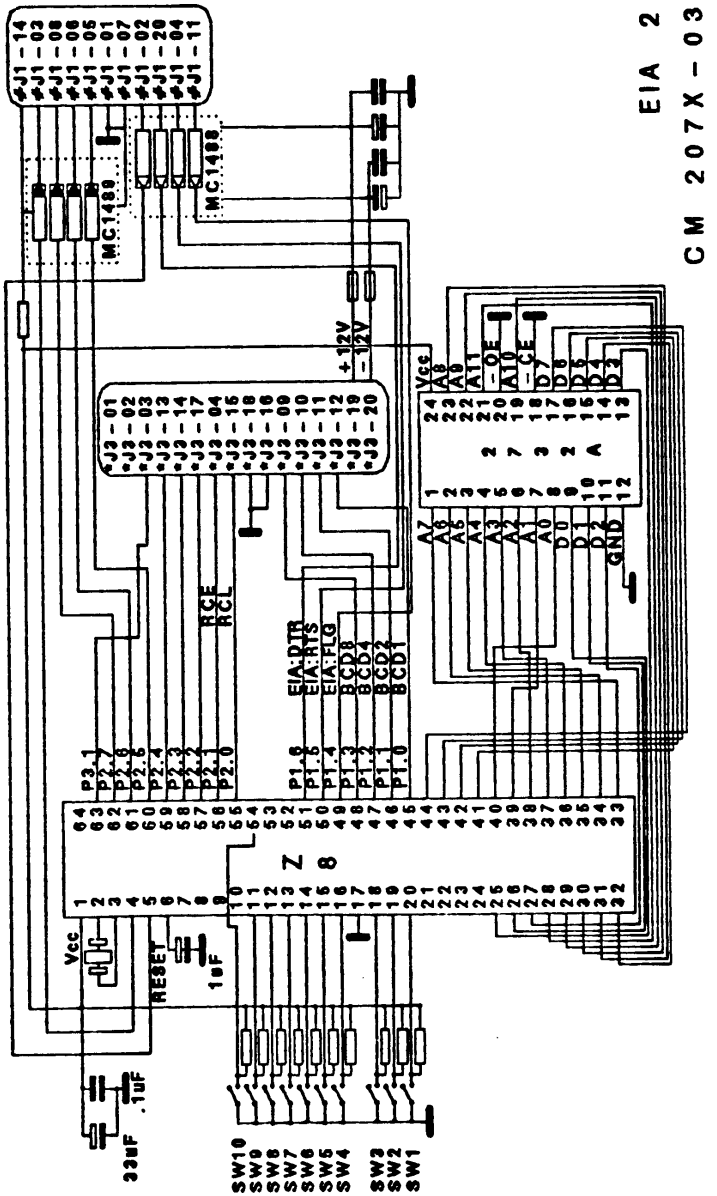
APPENDIX D

D. COMPONENT LAYOUT



APPENDIX E

E. SCHEMATIC



23-MAR-

Το interface 2071 βλ. CF 207x-03 σε οδηγί

1. Είναι παγωμένο σε 8-bit NO PARITY 2S
2. Έχει ρυθμισθεί σε 9600 Baudrat
3. Η count.lib διαφέρει εντελώς στο module da
   στοραδμό που διαβάζει με μιας string 18char
   ~~DC1~~ ÷
   DC1-DC3 – Count 8chars – Time 8chars
4. Το πρόγραμμα control ελέγχει υβριδικά τον counte
   τους χαρακτήρες: "S" start, "P" stop, "T" status, "R"
5. Ένας κύκλος Reset – Start – έλεγχος Stop από τη μο
   Read-Out έχει ως εξής:
   S reset/start
   T status: περιμένω να πάρω 17 (DC1) αμέσως
   T status: παίρνω 19 (DC3) όταν τελειώσει η μέτρηση
   R read: πρέπει να λάβω στην αρχή DC1, DC3 και κ
   counts και time

6. Με βάση το πιο πάνω σκεπτικό πρέπει να γίνουν τ
   τα προγράμματα εφαρμογής.

7. Το cable του παγοδιαιρέτη είναι εκείνο της
   6η. 24 (α/α 3) του παρόντος manual.

```
      program  control
c     Control Programme for CI Model 207x-03 EIA interface
c     BR 9600, 8-no parity- 2 stop : standard configuration
      character*1 bel,sta,stp,rea,ctrl,dc1,dc3,xon,xoff
      character*18 buffer
      integer*2 oct,sp
      data sp,ipn/231,1/
      bel=char(07)
      sta=char(24)
      stp=char(25)
      rea=char(23)
      xon=char(17)
      xoff=char(19)
      call setser(ipn,sp)
3     write (*,1)
1     format(1x,'Give Command Character (Start, stoP, sTatus, Read): '
     1 \)
      read (*,2)n
2     format(a1)
      if(n.eq.'S'.or.n.eq.'s')go to 10
      if(n.eq.'P'.or.n.eq.'p')go to 11
      if(n.eq.'T'.or.n.eq.'t')go to 12
      if(n.eq.'R'.or.n.eq.'r')go to 13
      write(*,4)bel
      go to 3
4     format(1x,a1,'Invalid command character')
10    call tocnt(sta)
      go to 3
11    call tocnt(stp)
      go to 3
12    call fmcnt(ctrl)
      i=ichar(ctrl)
      write(*,5)i
5     format(1x,'Character received from 2071 : ',i4)
      if(i.eq.xoff)write(*,6)
6     format(' Xoff received')
      go to 3
13    call tocnt(rea)
      call datain(ierr,buffer)
      write(*,7)ierr,buffer
7     format(' Readout status : ',i3,4x,'String received : ',a6)
      do 9 i=1,18
9     write(*,8)ichar(buffer(i:i))
8     format(1x,i5)
      if(ierr.ne.0)then
        write(*,16)bel
        go to 3
      endif
16    format(1x,a1,'Error during readout')
      read(buffer,17)dc1,dc3,icount,itim
17    format(2a1,2i8)
      write(*,20)itim,icount
20    format(1x,'Time : ',i8,10x,'Counts : ',i8)
      if(xon.ne.dc1)write(*,18)bell,ichar(dc1)
      if(xoff.ne.dc3)write(*,19)bell,ichar(dc3)
      go to 3
18    format(1x,a1,'Error during readout, xon  : ',i3)
19    format(1x,a1,'Error during readout, xoff : ',i3)
      end
```

```
        mov     sp,bp
        pop     bp
        ret     08h
err:
        cmp     al,17           ;DC1
        jne     contd
        cmp     al,21           ;DC3
        jne     contd
        cmp     al,32           ;Space
        jne     contd
        inc     ah              ;increase error indicator
        jmp     contd
datain  endp

rkbd    proc    near
        push    ax
        mov     ah,0bh
        int     21h
        cmp     al,0ffh
        jne     arkbd
        mov     ah,08h
        int     21h
arkbd:
        pop     ax
        ret
rkbd    endp
code    ends
end
```

```
fmcnt   proc far
        push    bp
        mov     bp,sp
        mov     dx,LSR
waitf:
        call    rkbd
        in      al,dx
        test    al,1h
        jz      waitf
        mov     dx,RBR
        in      al,dx
        les     bx,dword ptr [bp+6]
        mov     es:[bx],al
        mov     sp,bp
        pop     bp
        ret     04h
fmcnt   endp

wfmcnt  proc far
        push    bp
        mov     bp,sp
        mov     dx,LSR
        in      al,dx
        test    al,1h
        jz      nwait
        mov     dx,RBR
        in      al,dx
contw:
        les     bx,dword ptr [bp+6]
        mov     es:[bx],al
        mov     sp,bp
        pop     bp
        ret     04h
nwait:
        mov     al,00h
        jmp     contw
wfmcnt  endp

datain  proc far
        push    bp
        mov     bp,sp
        mov     ah,00h
        les     bx,dword ptr [bp+6]
        mov     cx,18
ga:
        mov     dx,LSR          ;Test if
waitd:
        call    rkbd
        in      al,dx           ;reception
        test    al,1h           ;has been
        jz      waitd           ;completed
        mov     dx,RBR
        in      al,dx           ;Read character from rbr
        cmp     al,'0'          ;Character
        jl      err             ;must be
        cmp     al,'9'          ;between
        jg      err             ;"0" and "9"
contd:
        mov     es:[bx],al      ;Put character into string
        inc     bx              ;Increase string address
        loop    ga
        les     bx,dword ptr [bp+10]
        mov     es:[bx],ah
```

```
name communicate_with_2071
;AUTHOR : Stavros Vossiniotis    May 1988 ; 2071 interface 23-03-97
data    segment public 'data'
dtbr    dw      03f8h
        dw      02f8h
        dw      03e8h
        dw      02e8h
dlsr    dw      03fdh
        dw      02fdh
        dw      03edh
        dw      02edh
two     db      02h
TBR     dw      ?
RBR     dw      ?
LSR     dw      ?
data    ends
dgroup  group   data

code    segment 'code'
        assume  cs:code,ds:dgroup,ss:dgroup
public  setser,tocnt,fmcnt,wfmcnt,datain

setser  proc    far
        push    bp
        mov     bp,sp
        les     bx,dword ptr [bp+10]
        mov     dx,es:[bx]
        mov     ax,dx
        mul     two
        mov     ah,00h
        mov     di,ax
        mov     bx,dtbr[di]
        mov     TBR,bx
        mov     RBR,bx
        mov     bx,dlsr[di]
        mov     LSR,bx
        les     bx,dword ptr [bp+6]
        mov     ax,es:[bx]
        mov     ah,00h
        int     14h
        mov     ax,3301h
        mov     dl,01h
        int     21h
        mov     sp,bp
        pop     bp
        ret     08h
setser  endp

tocnt   proc    far
        push    bp


waitt:
        in      al,dx
        test    al,20h                  ;Wait till TBR is empty
        jz      waitt
        mov     dx,TBR
        les     bx,dword ptr [bp+6]
        mov     al,es:[bx]              ;Read character parameter
        out     dx,al                   ;Transmit it
        mov     sp,bp
        pop     bp
        ret     04h
tocnt   endp
```