

Lucrarea 3

INTERFAȚA SERIALĂ RS-232C. REGISTRELE DE DATE, STARE ȘI CONTROL

3.1. Instalarea și configurarea interfețelor seriale.

Se consideră o situație în care se dorește interconectarea a două calculatoare personale, adică o conexiune **DTE-DTE**, prin cablu „**null-modem**”. Pentru aceasta trebuie ca parametrii de comunicație, ai interfețelor, să fie aleși la fel pe cele două terminale.

Configurarea unei interfețe seriale în cazul calculatoarelor personale pe care operează sisteme Microsoft Windows.

La instalarea sistemelor de operare din familia Microsoft Windows, interfețele seriale standard sunt recunoscute și instalate automat. Totuși, există și excepții care vor trebui tratate individual, conform manualului de instalare și utilizare.

În funcție de versiunea sistemului de operare, configurația și formatul meniului și al ferestrelor poate să fie diferit. Pentru exemplificare s-a folosit interfața standard oferită de versiunea XP a acestei familii de sistem de operare.

Din meniul **Start/Control Panel** se alege **System/Hardware/Device Manager**, se caută în listă **Ports (COM & LPT)**, iar apoi se marchează portul serial corespunzător, așa cum se poate vedea în **figura 3.1**.

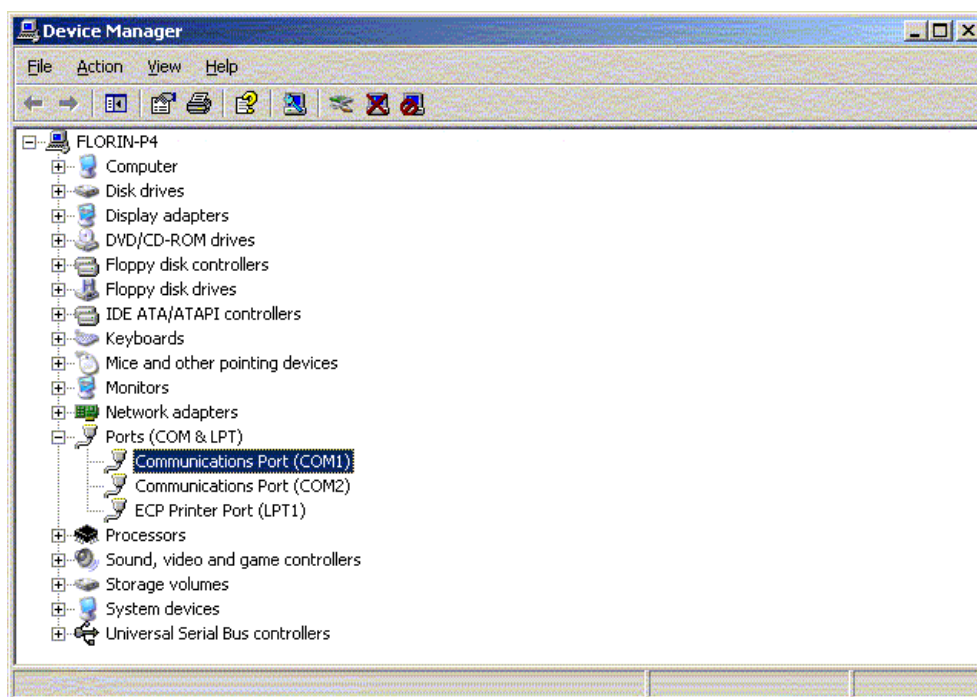


Fig. 3.1. Selectarea portului serial, din fereastra Device Manager.

Este necesar să se aleagă o rată de transfer, măsurată în **bți pe secundă – bps** (viteză de transfer), adaptată la lungimea cablului de legătură dintre **DTE** și **DCE** sau dintre două **DTE**.

Controlul fluxului depinde de tipul cablului și al echipamentelor folosite. În cazul utilizării unui cablu **null-modem**, prezentat în **tabelul 2.2** (din cursul nr. 2), la care sunt accesibile semnalele **RTS** și **CTS**, se va folosi controlul **hardware** al fluxului (*hardware flow control*), iar în cazul unui cablu cu trei fire, doar pentru **RxD**, **TxD** și **GND**, cel **software**, bazat pe caracterele de control **Xon** și **Xoff**,

transmise prin circuitele de comunicație **TxD** și **RxD**. Această soluție, deși aparent mai simplă, nu este recomandată datorită scăderii ratei de transmisie și a erorilor apreciabile de nesincronizare.

Deși formatul de serializare a datelor se poate modifica, cu condiția ca cele două terminale să fie configurate la fel, se recomandă cel implicit de **8 biți de date**, **1 bit de stop** și **nenulbit de paritate**. Formatul cu 7 biți de date și un bit de stop se folosește doar în cazul echipamentelor construite cu procesoare de 8 biți, din motive de optimizare a rutinelor de împachetare și serializare, așa cum se poate vedea în **figura 3.2**.

În cazul utilizării, pentru extinderea numărului de interfețe seriale, a unor componente suplimentare, poate fi necesară configurarea unor opțiuni avansate.

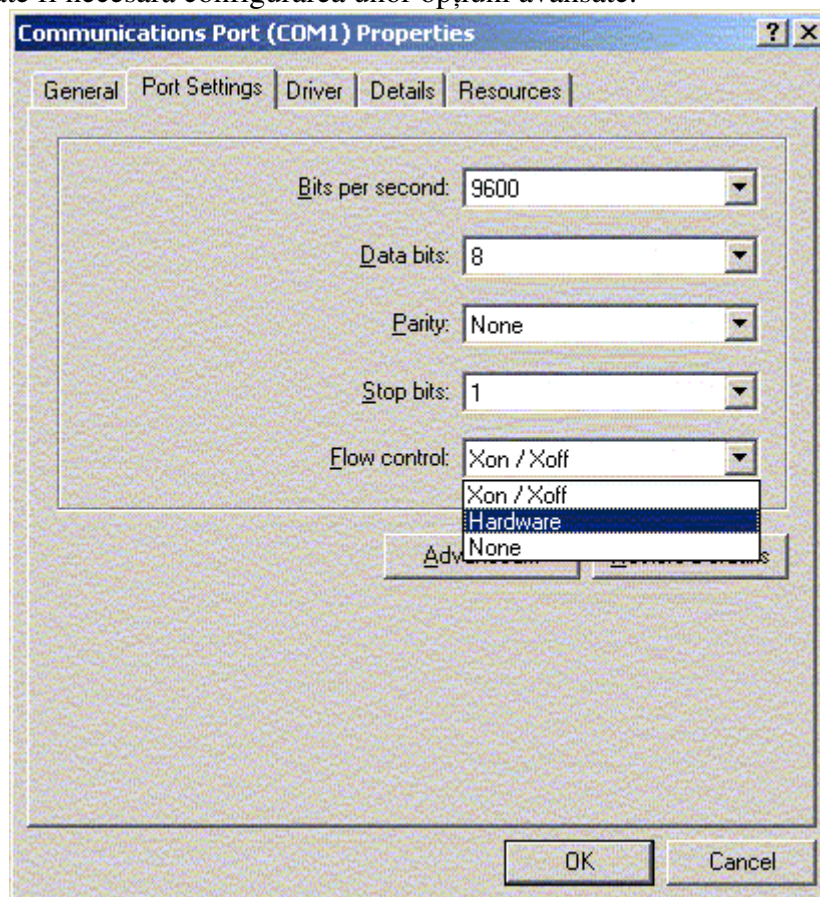


Fig. 3.2. Configurarea parametrilor de comunicație ai unui port serial.

Configurarea utilitarului Hyperterminal.

În afară de configurarea interfețelor seriale, trebuie configurate aplicațiile care gestionează transferul datelor. Acestea au denumirea generică de „**emulator de terminal**” și se găsesc într-o varietate mare de forme, unele dintre ele distribuite odată cu sistemele de operare.

Denumirea lor vine de la faptul că, operând pe un calculator personal, înlocuiesc funcțiile unui echipament video-terminal (console), folosite în trecut pe scară largă în rețelele **Main Frame** și **Unix**, ca terminale pentru echipamente industriale dedicate, sau mașini cu comandă numerică.

În prezent, se folosesc la interfațarea unor echipamente de comunicație, sau a sistemelor microprogramate. Aplicațiile cel mai des întâlnite sunt prezentate în **tabelul 3.1**.

Tabelul 3.1.

Sistem de operare	Aplicație Terminal
DOS	Comit, Procomm, Telix, Term95
MS Windows	Hyperterminal, PuTTY, BitCom
Unix, Linux	Minicom, xterm, rxvt

Aceste aplicații seamănă mult între ele, opțiunile de configurare fiind aceleași. Deoarece folosesc același standard de transfer, ele se pot interconecta direct fără să existe riscul de incompatibilitate.

Hyperterminal este un utilitar furnizat odată cu sistemul de operare. Configurarea aplicației se face după lansarea ei în execuție care se găsește în grupul **Accessories / Communications**. Se lansează **Hypertrm.exe** care deschide o fereastră de interogare, pentru un nume de conexiune și o pictogramă, așa cum se poate vedea în **figura 3.3**.

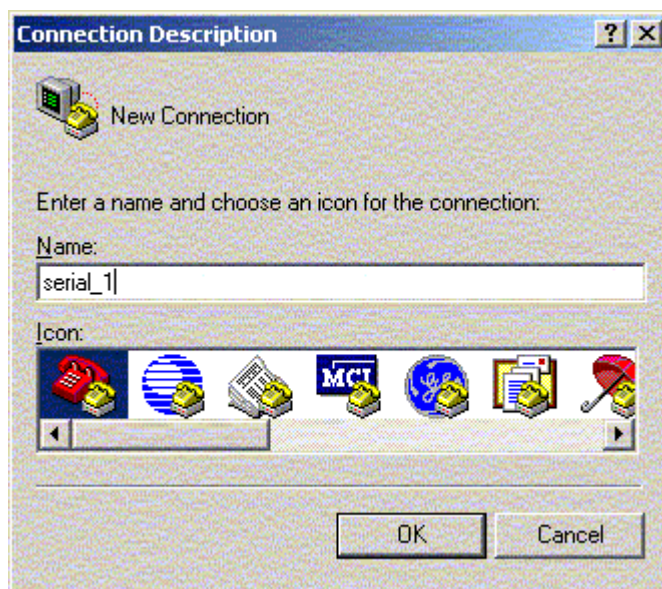


Fig. 3.3. Crearea unei sesiuni de conectare.

Utilizând aceleași opțiuni pentru fiecare dintre cele două terminale (calculatoare) conectate prin cablu „**null-modem**”, cu ajutorul aplicațiilor emulatoare de terminal, se pot transfera mesaje text sau fișiere.

În cazul interconectării unor terminale care utilizează sisteme de operare diferite, la transferul mesajelor text trebuie să se țină seama de faptul că acestea pot trata diferit combinația de caractere **LF** (*Line Feed* – avansul liniei) și **CR** (*Carriage Return* – „întoarcerea carului” = sfârșitul unui rând).

Aceste două caractere, diferite între ele, sunt folosite de unele sisteme de operare împreună **CR+LF**, adică se presupune că, în general, după sfârșitul unui rând (linie) urmează rândul următor. În acest fel, cursorul este trimis automat la începutul rândului următor.

În cazul transmiterii caracterului **CR**, în absența lui **LF**, aplicația terminal îl poate genera automat cu scopul așezării în ordine a mesajelor text.

Protocoloale de modem.

Pentru transferul fișierelor se folosesc unele componente numite „protocoloale de modem”, care împart fișierele în blocuri și le transferă folosind mecanisme de verificare și control al erorilor. Cele mai cunoscute dintre acestea sunt:

- **ASCII**, transfer la nivel de șir de caractere ASCII,
- **XMODEM**, **YMODEM**, **ZMODEM**, transfer prin segmentare în blocuri cu dimensiune de **128 octeți**, sau de **1024**, cu mecanisme de control a erorilor, specifice funcțiilor de transport. Variantele „-G” folosesc transfer în șir continuu, mai rapide dar cu risc ridicat de apariție a erorilor. **ZMODEM** este cel mai performant, având mecanisme evolute de transport. Transferul blocurilor se face cu confirmare, cu fereastră cu dimensiune variabilă și funcții de pornire automată a transferului între terminale. A fost implementat în multe aplicații, sub aproape toate sistemele de operare contemporane lui. Succesul implementării în sistemele de operare Unix și Linux au făcut ca acesta să le înlocuiască pe primele.

- **Kermit** este o colecție de aplicații asociate unui protocol de transfer de date și fișiere, conceput pentru a permite transferul datelor între diferite tipuri de terminale și calculatoare din familia IBM „main frame”. A fost dezvoltat să fie portabil, pe diferite familii de sisteme de operare, componente ale sale fiind folosite în prezent în sistemele cu microcontrolere.

După configurarea parametrilor de transfer, trebuie configurată interfața folosită de operator, în funcție de sistemul cu care se interconectează terminalul. Pentru interconectarea a două calculatoare personale se recomandă opțiunile **ANSI** sau **TTY**.

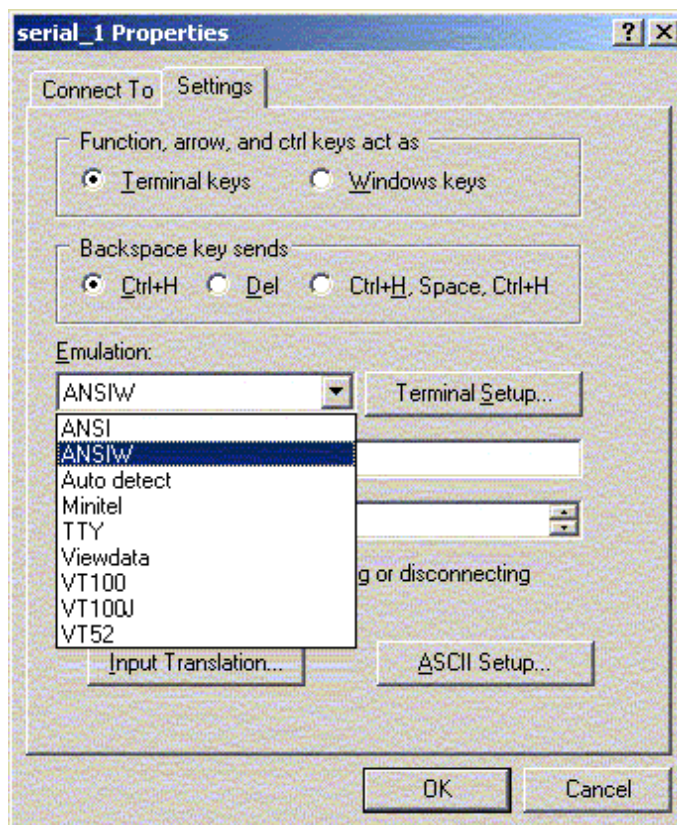


Fig. 3.4. Configurarea interfeței de lucru și a modului de afișare a datelor.

Modurile de afișare a datelor au fost create pentru a fi compatibile cu sistemele de teleimprimatoare, înlocuite azi de sisteme moderne de comunicație. Totuși, unele dintre aceste standarde sunt încă folosite de echipamente industriale, sau de comunicație, controlate prin terminale.

3.2. Configurarea posturilor seriale la sistemele de operare Linux.

Denumirea porturilor standard.

Sistemele de operare folosesc denumiri diferite pentru identificarea perifericelor. De asemenea, modurile de adresare pot fi diferite pentru intrare sau ieșire [19].

Sistemul de operare **DOS** și **Microsoft Windows** folosește pentru identificarea porturilor seriale standard denumirea **COMx**: (COM – Communication port, “x”=numărul portului, “:”=indică faptul că este vorba despre un dispozitiv, nu un fișier). Pentru aceste sisteme de operare numele portului este același indiferent dacă este folosit ca intrare sau ieșire. Practic *buffer*-ul de intrare este diferit de cel de ieșire, dar acest lucru este mascat pentru utilizator. De exemplu **COM1**: folosește cererea de întrerupere **IRQ 4** (*Interrupt Request*) și are ca adresă **registrul** cu valoarea hexazecimală **3F8H** (0x3F8). Sistemele de operare din familia **Unix**, împreună cu cele derivate din acesta, **Solaris**, **Linux**, **BSD** etc., folosesc nume diferite pentru portul de intrare și ieșire. Astfel același port cunoscut în **DOS** și **Windows** sub indicativul **COM1**:, în **Linux** îl găsim ca

/dev/ttyS0 pentru portul de intrare și /dev/cua0 pentru cel de ieșire. În tabelul de mai jos se prezintă corespondența dintre denumirea porturilor în sistemele de operare **Linux** și **MS Windows**.

Linux Windows

ttyS0	(COM1)	adresa	0x3f8	IRQ	4
ttyS1	(COM2)	adresa	0x2f8	IRQ	3
ttyS2	(COM3)	adresa	0x3e8	IRQ	4
ttyS3	(COM4)	adresa	0x2e8	IRQ	3

Corespondența dintre porturile de intrare și cele de ieșire se poate vedea în liniile care urmează:

/dev/ttyS0 major 4, minor 64	/dev/cua0 major 5, minor 64
/dev/ttyS1 major 4, minor 65	/dev/cua1 major 5, minor 65
/dev/ttyS2 major 4, minor 66	/dev/cua2 major 5, minor 66
/dev/ttyS3 major 4, minor 67	/dev/cua3 major 5, minor 67

Se observă că aceste periferice se găsesc în structura de fișiere a sistemului de operare **Linux** în directorul corespunzător dispozitivelor, /dev/. Acest lucru se poate verifica cu ajutorul comenzii de listare `ls`:

```
[root@masserv /root]# ls -l /dev/cua*
crw-rw---- 1 root uucp 5, 64 Jan 11 11:46 /dev/cua0
crw-rw---- 1 root uucp 5, 65 Jan 11 11:46 /dev/cua1
...
[root@masserv /root]# ls -l /dev/ttyS*
crw-rw---- 1 root uucp 4, 64 Jan 11 11:46 /dev/ttyS0
crw-rw-r-- 1 uucp uucp 4, 65 Jan 11 11:46 /dev/ttyS1
...
```

Semnificația comenzii este:

- `ls` listează conținutul directorului /dev/ cu nume care începe cu `cua*` (* înlocuiește orice grup de caractere care),
- opțiunea `-l` definește modul în care se afișează lista (ca listă).

În cazul în care aceste periferice nu apar, în listă, pot exista câteva situații. La versiunile mai recente ale distribuțiilor de **Linux** s-a renunțat la separarea porturilor de intrare, de cele de ieșire. Astfel se poate întâmpla să nu apară nici un dispozitiv de tip /dev/cua*, dar să fie dispozitivele de tip ttyS*. În acest caz este necesară mai întâi o documentare asupra proprietăților specifice distribuției (adresare, opțiuni). O altă situație poate să apară în cazul în care la instalarea sistemului de operare **Linux** programul de instalare nu a generat aceste dispozitive, sau nu le-a găsit activate în **BIOS**. Dacă ele există și sunt activate, dar nu le găsim cu comenzile de mai sus, ele se pot genera cu comanda `mknod`:

```
[root@masserv /root]# mknod -m 666 /dev/cua0 c 5 64
[root@masserv /root]# mknod -m 666 /dev/ttyS0 c 4 64
```

Semnificația comenzii este:

- o `mknod` generează un nod de tipul /dev/cua0 sau /dev/ttyS0,
- o `-m` definește modul de acces (drepturile) al utilizatorilor la dispozitiv la valoarea 666, adică `rw-rw-rw-`, iar
- o `c` valoarea 5 64 (4 64) a numărului de periferic.

Pentru generarea dispozitivului `ttyS0` se poate utiliza și fișierul script `MAKEDEV` care se găsește în directorul `/dev/` folosind următoarea succesiune de comenzi:

```
[root@masserv /root]# cd /dev
[root@masserv /root]# ./MAKEDEV ttyS0
```

Acest fișier script se folosește pentru manipularea corectă a dispozitivelor de intrare și de ieșire, cu condiția ca cel care vrea să-l folosească să aibă drepturile necesare asupra directorului `/dev`.

Spre deosebire de sistemele Unix, în DOS și MS Windows aceste porturi sunt verificate la instalare și prezente în My Computer Properties... /Device Manager /Ports (COM & LPT) sau Start /Settings /Control Panel /System, așa cum se vede în **figura 2.4**. Modificarea opțiunilor se poate face acționând butonul Properties. De obicei sistemul de operare își alege acele opțiuni care conduc la o funcționare optimă a calculatorului. Aceasta se face automat prin alegerea parametrilor care evită conflictele și se recomandă păstrarea acestor setări. Opțiunile modificate manual (de către utilizator) vor fi păstrate și devin restricții suplimentare pentru sistemul de operare. Dacă totuși anumite situații impun schimbarea parametrilor, aceasta se poate face, având grijă să fie evitate conflictele de adrese, sau cereri de întrerupere.

Adăugarea unui adaptor serial multiport.

În cazul în care este necesară adăugarea unui adaptor serial multiport trebuie să se țină seama de resursele deja folosite ale calculatorului. Aceasta înseamnă că trebuie să se vadă mai întâi dacă sunt libere domeniile de adrese pe care le folosește adaptorul. De obicei aceste adaptoare folosesc una sau mai multe cereri de întrerupere pentru toate porturile seriale pe care le deservește. Aceste cereri de întrerupere trebuie să fie diferite de cele folosite de alte dispozitive. De exemplu, se poate folosi un adaptor cu 8 porturi care utilizează o singură cerere de întrerupere (de exemplu `IRQ 5`) și are 8 adrese distincte, separat pentru fiecare port. Astfel, adresele utilizate pentru cele 8 porturi disponibile prin adăugarea unui adaptor serial cu 8 porturi vor fi următoarele [19]:

<code>ttyS0</code>	(COM1)	adresa	<code>0x3f8</code>	<code>IRQ 5</code>
<code>ttyS1</code>	(COM2)	adresa	<code>0x2f8</code>	<code>IRQ 5</code>
<code>ttyS2</code>	(COM3)	adresa	<code>0x3e8</code>	<code>IRQ 5</code>
<code>ttyS3</code>	(COM4)	adresa	<code>0x2e8</code>	<code>IRQ 5</code>
<code>ttyS4</code>	(COM5)	adresa	<code>0x250</code>	<code>IRQ 5</code>
<code>ttyS5</code>	(COM6)	adresa	<code>0x258</code>	<code>IRQ 5</code>
<code>ttyS6</code>	(COM7)	adresa	<code>0x260</code>	<code>IRQ 5</code>
<code>ttyS7</code>	(COM8)	adresa	<code>0x268</code>	<code>IRQ 5</code>

În Windows, instalarea unui adaptor serial multiport depinde de tipul acestuia. În cazul unui adaptor implementat *hardware*, acesta este detectat, la pornirea calculatorului, de către programul de verificare și configurare al plăcii de bază (BIOS). În această situație, în baza cererii de întrerupere și a adreselor, care sunt active pe durata întreruperii respective, sistemul de operare alocă resursele în mod automat. Ceea ce trebuie adăugat este tipul de port asociat acestor adrese. De obicei, resursele *hardware* apar declarate în tabela cu configurația calculatorului, la pornirea acestuia. Dacă nu apar, sau dacă sistemul de operare Windows nu le detectează, atunci trebuie adăugate manual. Cu ajutorul meniului Start /Setting /Control Panel /Add New Hardware, se va adăuga un număr corespunzător de porturi seriale, atașate adreselor respective. Dacă adaptorul este implementat utilizând componente *software* suplimentare, instalarea acestuia se va face utilizând indicațiile și aplicațiile care însoțesc produsul.

În cazul sistemului de operare Linux, acestea trebuie adăugate manual, în lipsa unui utilitar care detectează automat un adaptor nou. Utilitarul `/sbin/setserial` are în componența fișierului de configurare `/etc/rc.d/rc.serial` suport pentru unele adaptoare mai des întâlnite. Se recomandă utilizarea unei versiuni actualizate a pachetului `setserial` pentru a avea suportul necesar pentru un

număr cât mai mare de adaptoare seriale. Trebuie avută în vedere compatibilitatea dintre acesta și versiunea nucleului (kernel și module). Dacă nu există suportul pentru instalarea automată a adaptorului se va folosi, pentru definirea logică a portului, următoarea secvență de comenzi [19]:

```
[root@masserv /root]# mknod -m 666 /dev/cua17 c 5 81
[root@masserv /root]# mknod -m 666 /dev/ttyS17 c 4 81
```

unde numărul minor de periferic “81” se obține din numărul portului “17” plus valoarea corespunzătoare a numărului minor “64” al primului port serial ttyS0 ($17+64=81$). În cazul utilizării scriptului MAKEDEV secvența de comenzi este următoarea:

```
[root@masserv /root]# cd /dev
[root@masserv /dev]# ./MAKEDEV ttyS17
...
[root@masserv /dev]# ls -l /dev/ttyS17
crw-rw---- 1 root uucp 4, 81 Jul 18 1994 /dev/ttyS17
[root@masserv /dev]# cd /root
[root@masserv /root]# .
...

```

Configurarea porturilor seriale se poate face cu utilitare, ca de exemplu setserial, apelate cu opțiuni adecvate, așa cum se poate vedea în exemplul de mai jos:

```
[root@masserv /root]# ./sbin/setserial /dev/ttyS2 port 0x3e8 irq 5
[root@masserv /root]# .
```

În urma apelării comenzii /sbin/setserial se asociază interfața /dev/ttyS2 registrului de la adresa 0x3E8 care va folosi cererea de întrerupere IRQ 5.

Această comandă se poate apela din fișiere script, interpretate automat la pornirea calculatorului, ca de exemplu /etc/rc.d/rc.serial, sau /etc/rc.d/rc.local. Acest fișier este citit în ultima etapă a procesului de inițializare, instrucțiunile conținute și executate putând modifica parametri fixați de o instrucțiune anterioară. Pentru adăugarea automată a portului de mai sus, acest fișier trebuie să conțină linia:

```
/sbin/setserial /dev/ttyS2 port 0x3e8 irq 5
```

Un alt fișier important, pentru configurarea interfețelor seriale, este /etc/serial.conf care conține, în mod centralizat, principalele opțiuni apelate apoi de către fișierele script. Pentru configurarea interfețelor seriale de bază, fișierul /etc/serial.conf trebuie să conțină următoarele linii:

```
# Acesta este un exemplu - serial.conf. Se poate elimina
# semnul # (comentariu) și face modificări potrivite
#
# Acestea sunt porturile standard COM1 - COM4
#
/dev/ttyS0 uart 16450 port 0x3F8 irq 4
/dev/ttyS1 uart 16450 port 0x2F8 irq 3
/dev/ttyS2 uart 16450 port 0x3E8 irq 5
#/dev/ttyS3 uart 16450 port 0x2E8 irq 3
#
```

2.3. Configurarea unei aplicații care să permită conectarea unui terminal, la un calculator pe care operează un sistem Linux.

Pentru conectarea unui alt terminal, pe linie serială, este nevoie de instalarea și pornirea unei aplicații care se numește generic `getty`. Această aplicație gestionează accesul prin interfețele seriale, generează interfața de comunicație de tip `tty` și lansează aplicații suplimentare necesare autentificării (*Remote Login*) și interpretării comenzilor (*Shell Interpreter*). Există mai multe aplicații care permit aceste funcții (`getty`, `agetty`, `mgetty`, `ugetty` etc.), în acest exemplu se va folosi `mgetty`, dintr-un pachet complex `mgetty+sendfax`, conceput pentru gestiunea funcțiilor de modem, de date, voce și fax. Această aplicație se poate instala odată cu sistemul, sau ulterior [19, 25]. Pentru a vedea dacă este prezentă pe sistem, se va folosi comanda `which`, astfel:

```
root@masserv:~# which mgetty
/sbin/mgetty
root@masserv:~#
```

Dacă aplicația este instalată, fișierele de configurare trebuie să se găsească în directorul `/etc/mgetty+sendfax`. Pentru a afla acest lucru, se poate folosi comanda `ls -l`, ca în exemplul de mai jos:

```
root@masserv:~# ls -l /etc/mgetty+sendfax
...
-rw----- 1 root root 2863 Nov 21 2006 login.config
-rw-r--r-- 1 root root 1849 Mar 15 2005 mgetty.config
root@masserv:~#
```

Deși acest director conține mai multe fișiere, utilizate de aplicația `mgetty`, pentru scopul prezentat în acest capitol este necesară editarea doar a celor două, listate mai sus.

Fișierul `/etc/mgetty+sendfax/mgetty.config` conține opțiuni pentru portul serial folosit, nivelul de înregistrare a evenimentelor și pentru tipul interfeței de afișare a mesajelor (tipul terminalului).

```
# Fișier de configurare mgetty
# liniile care încep cu "#" sunt comentarii,
# iar liniile goale sunt ignorate

# ----- setări globale -----
# debug reprezintă nivelul la care se face înregistrarea
# evenimentelor. 4 înseamnă înregistrarea accesului,
# utilizator, oră
debug 4
term vt100
# rata de transfer inițială 115200 bps
speed 115200

# ----- setări particulare -----
# Conexiune directă prin portul /dev/ttyS1 de pe un terminal
# VT100 care nu acceptă întreruperea semnalului DTR
# "toggle-dtr n(o)". "direct y(es)" înseamnă "fără modem"
port ttyS1
direct y
toggle-dtr n
# aceste opțiuni permit ca /dev/tty-port
```

```
# să fie în proprietatea "uucp.uucp"
# iar drepturile de acces să fie "rw-rw-r--" (0664).
port-owner uucp
port-group uucp
port-mode 0664
```

Fișierul `/etc/mgetty+sendfax/login.config` conține opțiuni necesare autentificării și lansării automate în execuție a unor aplicații, după conectarea unui utilizator și trebuie să conțină informații despre utilizator și procesul de autentificare.

```
# login.config
# Format:
#     username userid utmp_entry login_program [argumente]
# unde:
#     username = numele utilizatorilor care au drept de conectare
#     userid = identificatorul al utilizatorului în numele căruia operează
#             comanda. „-” înseamnă root
#     utmp_entry = numele care apare în fișierul de înregistrare a accesului
#                 se utilizează „@” pentru a marca numele de intrare
#     login_program = numele programului lansat automat, dacă se lansează login,
#                   autentificarea se face pe baza listei de utilizatori înregistrați în sistem.
#                   Pentru asta trebuie folosit „@” în câmpul de argumente
#     pentru acceptarea prin autentificare doar a utilizatorilor din sistem se vor
#     folosi opțiunile de mai jos:
#
#     nume userid utmp      program_login [argumente]
#
#     *      -      -      /bin/login @
#
```

Pentru lansarea automată a aplicației `mgetty`, la mai multe nivele de rulare (2, 3, 4, 5) ale sistemului de operare, se introduce în fișierul `/etc/inittab` următoarea linie (sau mai multe, dacă se dorește conectarea mai multor terminale seriale), așa cum se poate vedea în rândurile de mai jos.

```
#
# /etc/inittab      Acest fișier descrie modul în care
#                   decurge procesul de inițializare INIT
#                   pentru diferite nivele de rulare.
#
# .....
#
# Adăugarea unui port pentru consola serială:
s1:2345:respawn:/sbin/mgetty /dev/ttyS1 -s 115200 -r -x 4
#
# Semnificația parametrilor este următoarea:
# -s = rata de transfer este de 115200 bps,
# -r = operează pe linie serială directă, fără modem,
# -x = nivelul de înregistrare a accesului 4.
#
```

Înainte fazei de autentificare, aplicațiile din familia `getty` pot trimite un mesaj, sub forma unei pagini text (*logo*), cu informații de prezentare și identificare ale sistemului. Forma acestui mesaj se poate „personaliza” prin editarea conținutului fișierului `/etc/issue`, care poate să includă secvențe text, variabile ale sistemului, sau instrucțiuni de colorare a fundalului și textului. Trebuie remarcat faptul că se pot transmite doar mesaje text, șiruri constante sau variabile, fără figuri. Deoarece multe dintre aplicațiile de tip terminal permit utilizarea unor secvențe de „conversație” (*chat scripts*), informațiile conținute de aceste mesaje permit transferul unor informații care permit automatizarea procesului de conectare și autentificare.

În cazul instalării unui sistem care deservește un departament, mesajul de întâmpinare poate să conțină informații despre acesta, cum ar fi numele departamentului, versiunea sistemului de

operare, numărul de utilizatori conectați, sau linia pe care se conectează utilizatorul curent, așa cum se poate vedea în exemplul prezentat mai jos.



Fig. 3.5. Fereastra aplicației Hyperterminal, după conectarea, prin linie serială, la un sistem Linux.

Pentru a obține mesajul prezentat în **figura 2.8**, fișierul `/etc/issue` trebuie să conțină următoarele linii de cod:

```
-----
Universitatea Tehnica din Cluj-Napoca
Catedra de Masurari Electrice

      M      E
Cluj-Napoca  MMM  EEE
              .:MMMM:..:EEEE:..
----- \r
Masina: \m * Sistem: @ \r
Data: \D      Ora: \T
Utilizatori conectati: \U
Linia: \L
```

- Semnificația principalelor variabile, introduse alături de textul propriu-zis, este următoarea:
- @ - numele simbolic al sistemului (terminalului);
- \n - avans la linie nouă (*line feed*);
- \r - avans la început de rând (*carriage return*);
- \g - avertizor sonor (bell - clopoțel);
- \b - șterge caracterul anterior (*backspace*);
- \f - avans la o pagină nouă (*form feed*);
- \t - caracterul TAB;
- \s - denumirea sistemului de operare;
- \m - tipul platformei hardware;

- \V \R - versiunea, respectiv ediția sistemului de operare;
- \P \L - numele portului serial pe care se face conectarea;
- \N \U - numărul de utilizatori conectați la terminal;
- \S - rata de transfer la care s-a realizat conectarea;
- \D - data curentă în formatul ZZ/LL/AA;
- \T - ora curentă în formatul OO:MM:SS;
- \\ - caracterul \

NOTĂ: Deoarece fiecare aplicație de tip `getty` poate avea propriile opțiuni și parametri de configurare, trebuie să se țină cont de următoarele:

1. Trebuie studiată documentația, specifică fiecărei aplicații folosite, sintaxa putând fi diferită de aceasta, putând exista și variabile care permit modificarea culorii fondului, sau a caracterelor.

2. Aplicația `Login` poate să afișeze pe ecran, în timpul procesului de autentificare, propriile mesaje.

Cele prezentate în acest paragraf sunt valabile și în cazul utilizării unui echipament **DCE** (modem) conectat la o interfață serială. În acest caz trebuie modificate și adăugate anumite opțiuni specifice controlului funcțiunilor modemului, care trebuie privit doar ca o „prelungire” funcțională a interfeței seriale.