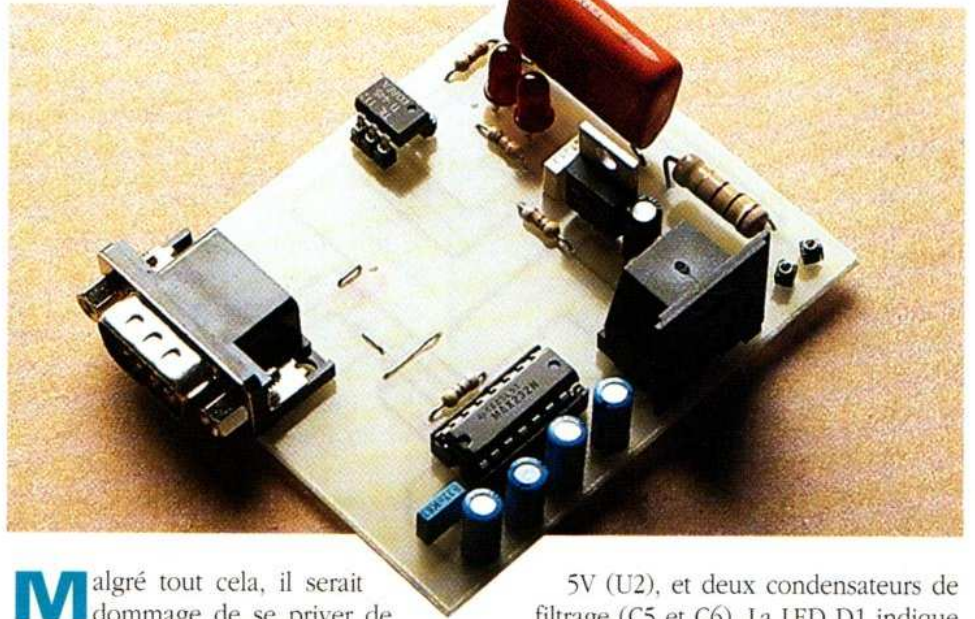


# Interface PC - Minitel

## pour micro serveur télématique

Avoir son propre serveur télématique peut se révéler d'une très grande utilité : service de renseignements pour une association, catalogue pour une entreprise, surveillance à distance de la maison, serveur à thèmes, etc. Ces serveurs personnels, appelés RTC par abus de langage (pour Réseau Téléphonique Commuté), ont connu un succès immense il y a quelques années. Monovoie la plupart du temps, ils étaient l'œuvre de passionnés et offraient de nombreux services dont les BALs (Boite Aux Lettres), rubriques, forums et téléchargement. Aujourd'hui, hélas, la folie du RTC s'est éteinte et cela en grande partie à cause de France Telecom. En effet, en augmentant le prix des communications locales, l'opérateur français a freiné l'ardeur des nombreux adorateurs du RTC. Heureusement, tous les serveurs n'ont pas disparu, mais les adeptes se font de plus en plus rares. De plus, avec l'arrivée massive des modems sur le marché, les 1200 bauds du Minitel s'essouffent et ne résistent pas aux 14400 bauds des BBS (Bulletin Board Service), serveurs en grande partie utilisés pour le téléchargement ou les accès Internet.



**M**algré tout cela, il serait dommage de se priver de l'occasion de posséder son propre serveur. Il suffit de disposer d'un Minitel, d'un PC, de l'interface dont nous vous proposons la réalisation ce mois-ci, et d'un compilateur C pour la partie informatique. Il n'y a absolument aucun frais à reverser à France Telecom (si ce n'est la location du Minitel) et les frais d'appel sont évidemment ceux d'une communication normale.

### L'interface PC-Minitel

L'interface que nous vous proposons se réduit à sa plus simple expression (figure 1) : elle possède un détecteur de sonnerie, une alimentation, et un adaptateur de niveau pour la liaison série. Pour ce qui est des spécificités du Minitel et de l'interface série, veuillez vous reporter aux pages précédentes.

Tous les composants sont standards et le coût total ne devrait pas excéder 70F.

### L'alimentation

Grâce à la sortie 8,5 V 1 A du Minitel, l'alimentation ne nécessite qu'un régulateur

5V (U2), et deux condensateurs de filtrage (C5 et C6). La LED D1 indique la mise sous tension de l'interface.

### Le module détection sonnerie

Les fils 3 et 5 du téléphone sont amenés sur CON2. Lors d'une sonnerie, la ligne envoie un signal sinusoïdal à 50 Hz dont l'amplitude est d'environ 80 V crête à crête. Au repos, seule une tension continue de 48 V est visible.

La capacité C8 filtre la tension continue et ne laisse donc passer le courant que lors d'une sonnerie. La zener D2 ainsi que la diode D3 redressent la tension et la fixent à 9,1 V (R3 permet de faire chuter la tension). C9 transforme le signal sinusoïdal en signal continu, allumant ainsi la LED D4 et actionnant le photocoupleur U3. Ainsi, la broche RI de la prise série (CON3) est à + 5 V lors d'une salve de sonnerie, et à 0V le reste du temps.

### L'adaptation des signaux série

Cette fonction est laissée aux bons soins du circuit MAX232 (U1), spécialisé en la matière. Comme TX du Minitel est à col-



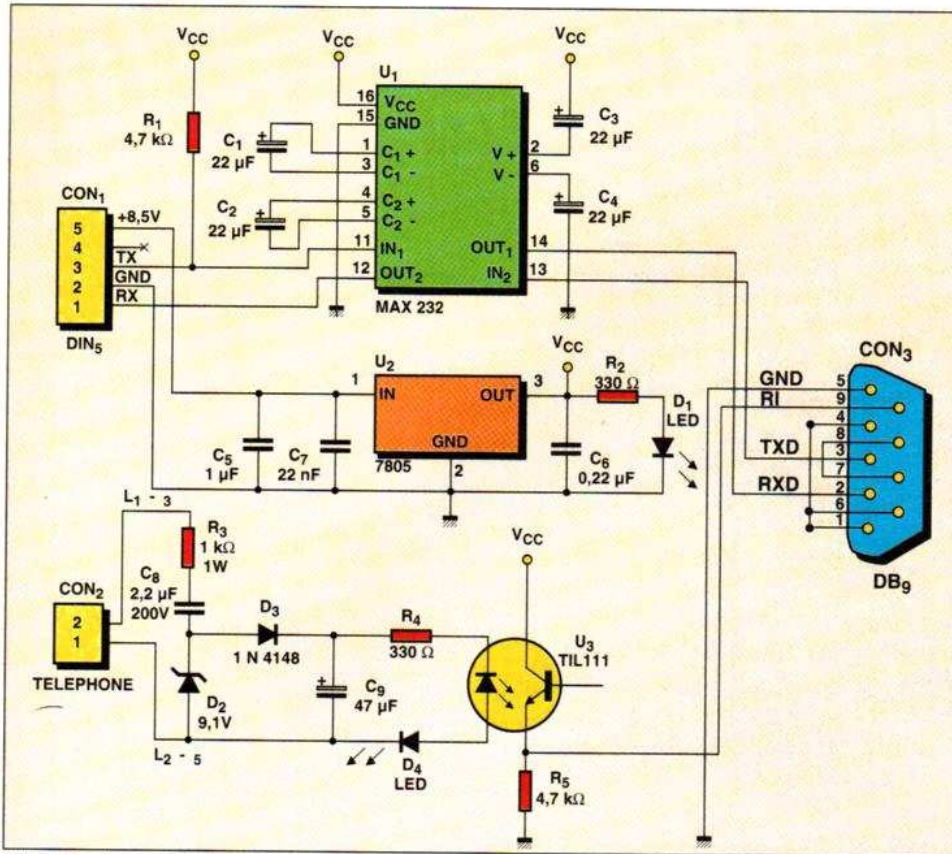


Figure 1 - Schéma de l'interface PC/Minitel

## NOMENCLATURE

- R1 4,7 kΩ
- R2, R4 330 Ω
- R3 1 kΩ 1 W
- R5 4,7 kΩ
- C1, C2, C3, C4 22 μF, radial, 63 V
- C5 1 μF, radial, 63 V
- C6 220 nF, plastique
- C7 22 nF, plastique
- C8 2,2 μF, non polarisée, 200 V
- C9 47 μF, radial, 63 V
- D1, D4 LED
- D2 zener 9,1 V
- D3 1N4148
- U1 MAX232 + support 16
- U2 7805
- U3 TIL111 + support 6
- CON1 prise DIN 5 broches 180° femelle pour CI
- CON2 bornier
- CON3 prise DB9 mâle soudée, pour CI

lecteur ouvert, une résistance de rappel est mise (R1). Les condensateurs C1 à C4 servent comme pompes de charge, permettant la création du 12 V.

## Réalisation

La figure 2 vous donne le typon simple face à réaliser et la figure 3 l'implantation des composants.

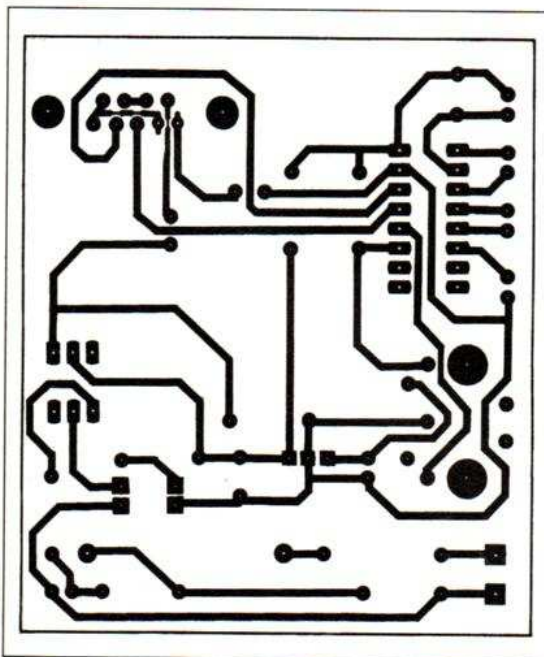


Figure 2 - Circuit imprimé, échelle 1, de l'interface PC/Minitel

Aucune remarque particulière n'est à faire quant à la réalisation si ce n'est de bien faire attention au sens des condensateurs C1 à C4.

## Programmation du serveur

La programmation d'un serveur télématique monovoie peut paraître compliquée

à première vue. Cependant vous verrez vite qu'un squelette répondant à vos besoins peut être mis en place assez rapidement et qu'à partir du moment où l'on s'affranchit de la gestion des BALs et des arborescences, les difficultés ne sont pas très nombreuses.

Le langage retenu a été le langage C. En effet, seul un langage évolué permettant à la fois une programmation système bas

niveau et une gestion facile de l'environnement PC garantit la puissance du serveur. Si vous ne le connaissez pas encore, cela vous fera une occasion de plus de vous y mettre.... Enfin, afin de ne pas être dépendant d'un compilateur spécifique, seules les fonctions de bases ont été utilisées. Le listing présenté vous donne les procédures bas-niveau pour la gestion du port série ainsi que quelques fonctions de gestion du Minitel (saisie d'une chaîne, affichage d'un message en ligne 0, etc....)

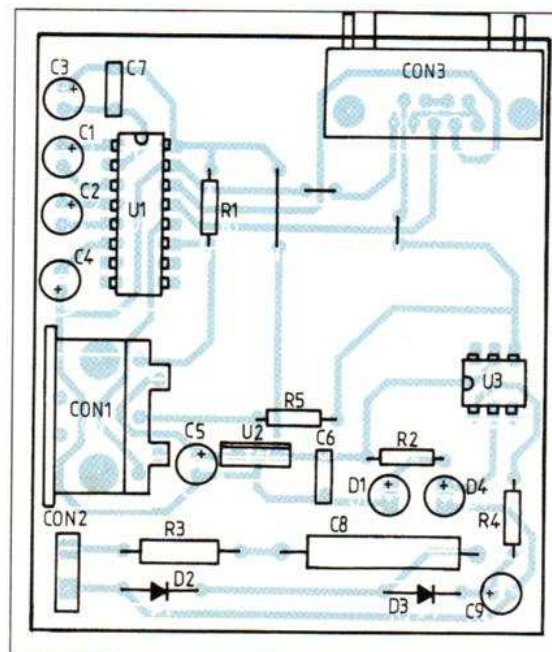


Figure 3 - Implantation des composants



## Listing du serveur de démonstration

```

#include <stdio.h>
#include <dos.h>

/*-----*/
/*          Section DATA          */
/*-----*/

/* Code des touches de fonction du Minitel */
#define ENVOI      0x41
#define SUITE     0x48
#define RETOUR    0x42
#define REPETITION 0x43
#define CORRECTION 0x47
#define ANNULATION 0x45
#define GUIDE     0x44
#define SOMMAIRE  0x46
#define CONNEXION 0x49

/* Numéro du port série utilisé */
#define COM 0x01 /* COM 2 (COM1 = 0x00) */

/*-----*/
/*          Procédures bas niveau          */
/*-----*/

/*-----*/
/* Initialisation du port série */
/*-----*/
void Init_Serial()
{
    union REGS pregs;
    /* fn 0: réglage du protocole */
    pregs.h.ah = 0x00;
    /* 7 bits, paire, 1200 bauds */
    pregs.h.al = 0x9A;
    pregs.x.dx = COM;
    int86(0x14, &pregs, &pregs); /* IT DOS 14 */
}

/*-----*/
/* Emission de caractères */
/* x: caractère à émettre */
/*-----*/
void Out(char x)
{
    union REGS pregs;
    pregs.h.ah = 0x01; /* fonction émission */
    pregs.x.dx = COM;
    pregs.h.al = x;
    int86(0x14, &pregs, &pregs); /* IT DOS 14 */
}

/*-----*/
/* Réception de caractères */
/* utilisation: car=Inp() */
/*-----*/
char Inp()
{
    union REGS pregs;
    pregs.h.ah = 0xff;
    while (pregs.h.ah & 0x80)
    {
        pregs.h.ah = 0x02; /* fonction réception */
        pregs.x.dx = COM;
        int86(0x14, &pregs, &pregs); /* IT DOS 14 */
    }
}

}
return (pregs.h.al);
}

/*-----*/
/* Caractere en attente ? */
/* si oui 1, sinon 0 */
/*-----*/
int Is_Inp()
{
    union REGS pregs;
    pregs.h.ah = 0x03; /* fonction tester état */
    pregs.x.dx = COM;
    int86(0x14, &pregs, &pregs); /* IT DOS 14 */
    if ((pregs.h.ah & 0x01) == 1)
        return (1);
    else
        return (0);
}

/*-----*/
/* Vide le buffer série */
/*-----*/
void Flush()
{
    union REGS pregs;
    pregs.h.ah = 0x00;
    while (!(pregs.h.ah & 0x80))
    {
        pregs.h.ah = 0x02; /* fonction réception */
        pregs.x.dx = COM;
        int86(0x14, &pregs, &pregs); /* IT DOS 14 */
    }
}

/*-----*/
/* Attente de la sonnerie du téléphone */
/*-----*/
void Wait_Ring()
{
    union REGS pregs;
    pregs.h.al = 0x00;
    while ((pregs.h.al & 64) != 64)
    {
        /* fonction tester état */
        pregs.h.ah = 0x03;
        pregs.x.dx = COM;
        int86(0x14, &pregs, &pregs); /* IT DOS 14 */
    }
}

/*-----*/
/*          Procédures de gestion du serveur          */
/*-----*/

/*-----*/
/* Envoi d'un message sur la ligne 0 du minitel */
/* c1: pointeur sur le titre du message */
/* c2: pointeur sur le message */
/*-----*/
void Sysmes(char *c1, char *c2)
{
    /* ligne 0 */
}

```



```

Out(0x1F);Out(0x40);Out(0x41);Out(0x18);
/* fond bleu */
Out(0x1B);Out(0x54);Out(' ');
while (*c1!=0) Out(*c1++);
/* fond rouge */
Out(' ');Out(0x1B);Out(27);Out(0x51);Out(' ');
while (*c2!=0) Out(*c2++);
Out(0x1B);Out(0x50); /* fond noir */
Out(0x0A); /* descend le curseur */
}

/*-----*/
/* Positionne le curseur en x y */
/*-----*/
void Pos(int x, int y)
{
    Out(0x1F);
    Out(0x40+y);
    Out(0x40+x);
}

/*-----*/
/* Envoi une chaine au minitel */
/* c: pointeur sur la chaine */
/*-----*/
void Send(char *c)
{
    while (*c!=0) Out(*c++);
}

/*-----*/
/* Saisie d'une chaine sur le minitel */
/* x,y: position de la saisie */
/* l : longueur de la chaine */
/* aff: si 1 affiche ce qui est tapé */
/*      sinon affiche '**' */
/* txt: chaine resultat */
/*-----*/
void Scan(int x, int y, int l, int aff, char *txt)
{
    int posx=x, posy=y, fin=0;
    int i, len=0;
    char c;
    Pos(posx, posy);
    for (i=0; i<l; i++) Out('.');
    Out(0x11); /* Fait apparaitre le curseur */
    Pos(posx, posy);
    while (fin==0)
    {
        c=Inp();
        if (c==0x13) /* touche de fn détectée */
        {
            c=Inp();
            if ((c==CORRECTION)&&(posx>x))
            {
                posx--;
                Out(0x08);
                Out('.');
                Out(0x08);
                txt[-len]='\0';
            }
            else if (c==ANNULATION)
            {
                txt[0]='\0';
                len=0;
                Pos(x,y);
                for (i=0; i<l; i++)
                Out('.');
                posx=x;
                Pos(posx, posy);
            }
            else if (c==ENVOI)

```

```

        fin=1;
    }
    else
    {
        if (posx-x+1<l)
        {
            posx++;
            txt[len++]=c;
        }
        else txt[len]=c;
        if (aff==1)
            Out(c);
        else
            Out('*');
    }
    Pos(posx, posy);
}
txt[len]='\0'; /* marque la fin de la chaine */
Out(0x14); /* fait disparaître le curseur */
}

/*-----*/
/* Attente de la connexion d'un hôte */
/* renvoie 1 si succès, 0 sinon */
/*-----*/
int Wait_Connect()
{
    long compteur;
    char c;
    Sysmes("System", "Attente de connexion");
    Wait_Ring();
    Sysmes("System", "Appel...");
    /* Opposabilité */
    Out(0x1B);Out(0x39);Out(0x6F);
    /* Connexion */
    Out(0x1B);Out(0x39);Out(0x68);
    /* Attente de la connexion de l'appelant */
    /* l'attente est interrompue si personne */
    /* ne répond au bout d'un temps donné */
    Wait_Connect:
    compteur=0;
    while (compteur<500000)
    {
        if (Is_Inp()==1)
            if (Inp()==0x13)
            {
                delay (100);
                if (Is_Inp()==1)
                {
                    c=Inp();
                    if (c==0x53)
                        return (1);
                    else if ((c==0x59)|| (c==0x50))
                        goto Wait_Connect;
                }
            }
        compteur++;
    }
    Sysmes("System", "Connexion annulée");
    Out(0x1B);Out(0x39);Out(0x67); /* Déconnexion */
    return (0);
}

/*-----*/
/* Boucle principale du serveur */
/*-----*/
void main()
{
    int local;
    char c, dummy[20];

```



```

Init_Serial();

Start:
local=0;
c=0;
Out(0x0C); /* Efface l'écran du Minitel */
Sysmes("PC", "Attente d'un evenement");
puts("\nServeur de démonstration\n");
puts("[c]onnexion locale");
puts("[a]ttente sonnerie");
puts("[f]in du programme");
scanf("%c",&c);
if (c=='a')
    while (Wait_Connect()==0);
else if (c=='c')
    local=1;
else if (c=='f')
{
    Out(0x0C);
    exit (0);
}
else goto Start;

Connect:
Out(0x0C); /* Efface l'écran */
Sysmes("System", "Connexion...");
if (local==0)
{
    /* prise vers modem */
    Out(0x1B);Out(0x39);Out(0x61);
    Out(0x5A);Out(0x53);
}

/* prise vers écran */
Out(0x1B);Out(0x3B);
Out(0x61);Out(0x58);Out(0x53);
/* annule clavier vers écran */
Out(0x1B);Out(0x3B);
Out(0x60);Out(0x58);Out(0x51);
/* annule modem vers écran */
Out(0x1B);Out(0x3B);
Out(0x60);Out(0x58);Out(0x52);

/* Ceci est un exemple de page */
Flush();
Out(0x0C);
Sysmes("System", "Bienvenue");
Pos(8,7);
Out(0x1B);Out(0x4D); /* Double hauteur */
Send("SERVEUR DE DEMONSTRATION");
Out(0x1B);Out(0x4C); /* Taille normale */
Pos(4,9);
Send("Auteur: E.L. pour le Haut Parleur");
Pos(3,17);
Send("Entrez votre pseudo:");
Scan(28,17,10,1,dummy);
printf("PSEUDO: %s\n",dummy);
Pos(3,19);
Send("Entrez votre password:");
Scan(28,19,10,0,dummy);
printf("PASSWORD: %s\n",dummy);
Sysmes("System", "A bientôt...");

/* Déconnexion */
if (local==0)
{
    Out(0x1B);Out(0x39);Out(0x67);
    while (Wait_Connect()==0);
    goto Connect;
}
else
    goto Start;
}
    
```

(c) Eric  
Larchevêque  
mai 1995

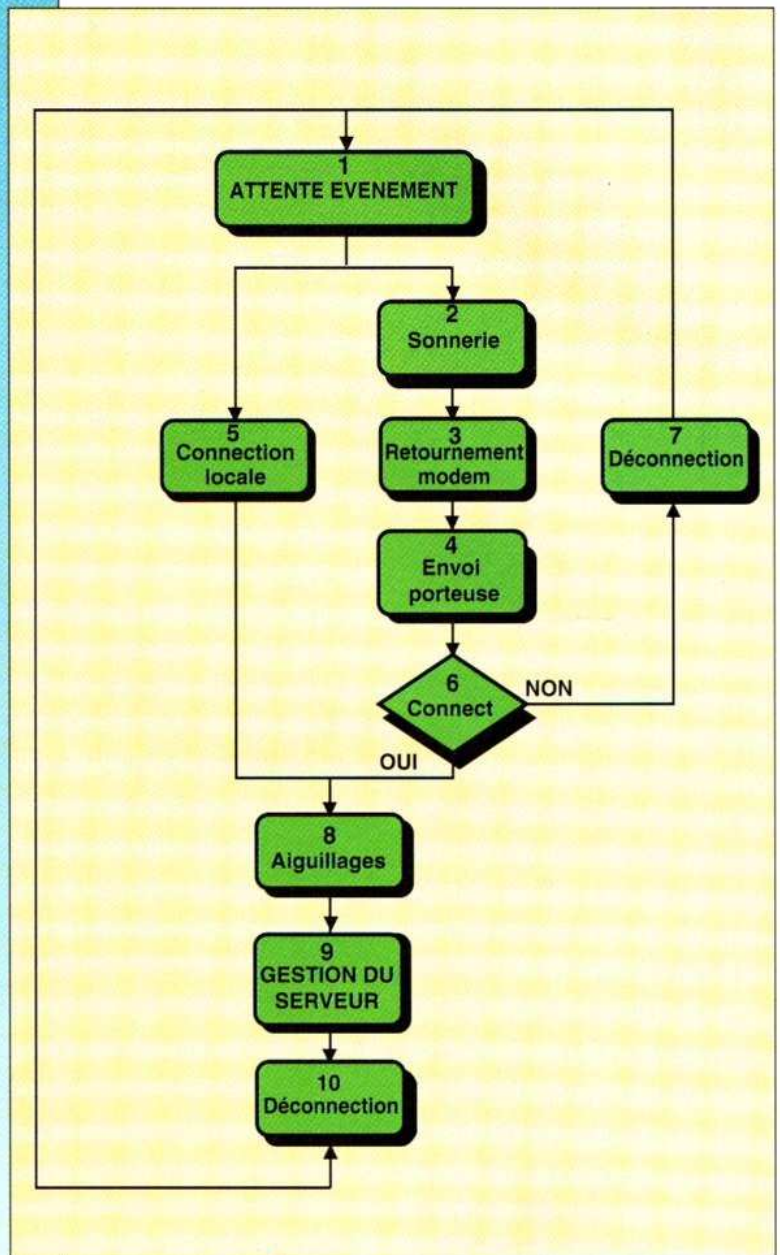


Figure 4  
organigramme du  
serveur.

Bien sûr, les routines de gestion du serveur n'ont pas été oubliées : détection sonnerie, connexion du modem, aiguillages, etc.

Une fois le programme compilé et lancé vous avez le choix entre attendre un appel et vous connecter en local. Le dernier choix vous permet de tester le serveur sans avoir à aller utiliser le Minitel du voisin.

Vous pourrez voir que l'exemple proposé est d'une rare simplicité.... A vous de le complexifier et de l'adapter à vos besoins.

Afin de mieux comprendre le fonctionnement du serveur, un organigramme résume les différentes étapes de l'appel à la déconnexion (voir figure 4).

E. Larchevêque / L. Lellu