



http://www.ci-industrielle.fr

Librairie DLL_BASE4

Logiciel	Librairie d'extension pour AS-VOX G4 - DLL_BASE4 - v1.16
Date de création	24/03/2000 - G. MISANDEAU - J. PALUD
Mise à jour	07/07/2003
	Copyright (c) 2000-2003 C.I.I. industrielle S.A. Cette documentation peut être distribuée et reproduite en respect des termes et conditions établies par l'Open Publication License v1.0 (accessible à http://www.opencontent.org/openpub/), C.I.I. industrielle - 8 rue Edgard Brandt - 72000 LE MANS - Tél : +33.243.500.600 - Fax : +33.243.500.601

La librairie dll_base4.dll contient diverses fonctions documentées ci-après. Cette documentation concerne les fonctions publiques de la librairie. La librairie dll_base4.dll peut être utilisée avec AS-VOX G4 sous Windows 2000.

BALEQUIV [v1.00]

Cette fonction recherche la boîte à utiliser en correspondance avec le numéro de boîte passé dans M4. Cette fonction sert à partager une boîte entre plusieurs postes (regroupement par service par exemple). Le numéro du poste (boîte demandée) est passé dans M4 en entrée de la fonction BALEQUIV qui détermine le numéro de la boîte à utiliser en fonction du fichier de routage **b_equiv.inf** (sous répertoire \g4_bal de l'installation). Ce fichier a le format suivant : **numposte numbalequiv**

Par exemple, si les postes 100, 102 et 104 doivent utiliser la boîte 104 le fichier **b_equiv** contiendra :

```
100 104
102 104
```

Si la boîte ne possède pas d'équivalence, son numéro est conservé. Dans l'exemple on constate que la boîte 104 n'a pas besoin d'être définie puisqu'elle n'est pas orientée vers une autre boîte.

Cette fonction peut être utilisée avant CKBALRET ou CKBALGEN pour gérer plus finement les boîtes vocales.

voir aussi : CKBALRET , CKBALGEN

CHECKCB [v1.00]

Cette fonction vérifie si le numéro de carte bancaire (clef de luhn) saisi dans le code d'accès de la rubrique appelante est valide. Si le numéro est valide, on réalise un saut sur la rubrique CARTEOK, sinon on saute sur la rubrique CARTEERR.

CKBALGEN [v1.00]

La fonction CKBALGEN vérifie que le numéro de la boîte vocale passée en paramètre dans le mémo M4 existe. Si la boîte est inexistante, le contenu de M4 est remplacé par le numéro de la boîte générale définie dans [SYSTEM] boitegen=xxxx dans le fichier PROVOXG4.INI (répertoire de Windows). Si la boîte désignée par M4 est existante, le retour se fait sans influence (l'arborescence continue normalement).

voir aussi : CKBALRET , BALEQUIV

CKBALRET [v1.00]

La fonction CKBALRET vérifie que le numéro de la boîte vocale passée en paramètre dans le mémo M4 existe. Si la boîte est inexistante, un retour est effectué dans l'application courante sur la rubrique BALERREUR. Si la boîte désignée par M4 est existante, le retour se fait sans influence (l'arborescence continue normalement).

voir aussi : CKBALGEN , BALEQUIV

CLIPM9 [v1.01]

Cette Fonction met le numéro de téléphone de l'appelant dans le memo M9. Si ce numéro n'est pas identifié, la fonction effectue un retour sur la rubrique nommée NO_CLIP. Cette fonction est utilisable uniquement sur des connexions numérique (T0 ou T2).

CODEUNIQUE [v1.00]

Cette fonction retourne dans M1 le numéro en correspondance avec un second numéro passé dans M2. Elle utilise un fichier au format : xxxxxxxxxxxx nnnn ou x peut être un numéro de client et n un numéro de poste cible.

Si la correspondance n'existe pas, un saut est effectué sur la rubrique CODEERR qui doit donc être accessible dans l'application courante.

CPTMAX [v1.00]

Cette fonction permet de vérifier combien de fois une personne est passée sur la rubrique qui fait appel à cette fonction lors d'une même communication. Elle utilise un fichier nommé *NomDeLaRubrique.cpt*. Ce fichier a le format suivant :

```
[ORIENTE]
ok=NomDeRubrique
stop = NomDeRubrique
memo=7
cptmax=3
```

cptmax désigne le nombre de fois maximum qu'il est possible de passer. *memo* désigne le numéro du mémo dans lequel on désire effectuer le comptage des passages (Ce mémo ne doit pas être utilisé par le reste de l'application).

ok désigne le nom de la rubrique sur laquelle on effectue un retour lorsque le nombre de passage est strictement inférieur à *cptmax*.

stop désigne le nom de la rubrique sur laquelle on effectue un retour lorsque le nombre de passage est égal à *cptmax*.

EQUIV [v1.00]

La fonction EQUIV recherche une valeur associée à une autre valeur dans une table d'équivalence. Exemple : La fonction EQUIV permet de retrouver le numéro de poste associé à un numéro de ligne directe.

La table se situe dans un fichier portant le nom de la rubrique faisant appel à cette DLL et porte l'extension eqv. Ce fichier doit se trouver dans le répertoire G4_dat. Le chemin d'accès à ce répertoire est défini dans Provox4.ini, section [SYSTEM], champ path_dat (En général : c:\g4\g4_dat).

Voici la forme générale d'un tel fichier :

```
[ERREUR]
retour=
MEMO=40
[GENERAL]
memo_in=1
memo_out=2
[EQUIV]
0243500600=30
0243500601=64
0243500651=25
.....
```

retour : permet de spécifier le nom de la rubrique sur laquelle on effectue un retour dans le cas où la correspondance n'existe pas. (Ici, on n'a pas spécifié de rubrique de retour).

memo : permet de spécifier une valeur de retour par défaut dans le cas où la correspondance n'existe pas. (Ici, la fonction renvoie 40 dans M2 si le numéro entré dans M1 ne se trouve pas dans table d'équivalence).

Remarque : on ne peut utiliser qu'un seul des deux paramètres à la fois. Un des deux doit toujours se trouver à 'blanc'.

memo_in : permet de spécifier le numéro de mémo par lequel on pourra lire la valeur dont on recherche l'équivalence. (Ici, il s'agit de M1).

memo_out : permet de spécifier le numéro de mémo par lequel on désire retourner la valeur équivalente. (Ici, il s'agit de M2).

La section EQUIV désigne la table d'équivalence.

GET_MCDU_M2 [v1.02]

Cette fonction récupère le MCDU (numéro abonné sollicité) de l'appel et le sauvegarde dans M2. Cette fonction est utilisable sur une connexion analogique avec protocole Q23 ou VPS.

HEURE [v1.00]

La fonction HEURE diffuse un message en fonction de l'heure système courante. Elle repose sur un fichier de planning C:\G4\G4_DAT\PLAGE.TXT au format suivant :

```
hhmm hhmm msg
```

par exemple :

```
0000 1200 C :MSGAV_MIDI.WAV
1200 2400 C :MSGAV_MINUIT.WAV
```

Les champs doivent être séparés par des espaces, et toutes les plages horaires doivent être couvertes.

LIREMSG1 .. LIREMSG4 [v1.00]

Cette fonction diffuse le message vocal correspondant au contenu du champ mémo M1 (pour LIREMSG1, M2 pour LIREMSG2...). Le fichier diffusé prend pour nom <Valeur de M1>.WAV et doit se trouver dans le répertoire g4_dat. Par exemple, si M1 est défini 000, le fichier g4_dat\000.WAV est diffusé lors de l'appel de la fonction LIREMSG1. En cas d'absence du message, aucune erreur n'est générée ni flaggée.

LOADAPP [v1.00]

Cette fonction permet de sélectionner une application sur les 4 premières voies du système en fonction du code d'accès à la rubrique appelante. Le fichier c:\g4\g4_dat\loadapp.txt doit contenir les codes et les applications correspondantes : par exemple

```
0000 appli0.apx
0001 appli1.apx
2000 thisyear.apx
```

Si le code ne correspond à aucune application, l'application courante n'est pas changée. La longueur du code d'accès doit correspondre à celle définie dans le fichier LOADAPP.TXT. Tous les codes ont donc la même longueur. Comme lors d'une configuration manuelle des applications par voie, la mise à jour des mémos ne se fait pas en direct. Les applications utilisées doivent avoir un nom SANS ESPACE.

MCHARGEMEMOS [v1.00]

La fonction MCHARGEMEMOS utilise le fichier de données portant le nom de la rubrique suivi de l'extension .txt. Elle permet de charger avec des valeurs définies dans ce fichiers les mémos M1 à M9. Ce fichier doit se situer dans le répertoire c:\spx_dat du serveur vocal. Le fichier possède un format type "fichier INI", soit une section MEMOS pouvant contenir les champs M1 à M9. Seuls les champs définis dans ce fichier sont chargés, les autres sont inchangés.

Exemple : mise dans M4 de la valeur 99 et dans M7 de la valeur 1220

[MEMOS] [v1.00]

```
M4=99
M7=1220
```

MCODERET [v1.00]

MCODERET est une fonction qui permet d'orienter le déroulement de l'appel vers une étiquette particulière en fonction d'un numéro donné par l'interlocuteur. CODERET utilise le fichier C:\G4\G4_DAT\nom_rubrique.RET pour orienter le déroulement de l'appel vers une rubrique en fonction du numéro donné. La rubrique faisant appel à MCODERET doit avoir un code d'accès multidigit (autant de '?' que de chiffres attendus, soit ????? pour 5 chiffres attendus). Si plusieurs applications utilisent MCODERET, il faut vérifier que les noms de rubriques sont différents (ou identiques s'ils utilisent les mêmes tables.)

Format du fichier <nom_rubrique>.RET :

```
xxx yyy RRRR avec
```

xxx = début de la plage yyy = fin de la plage RRRR = nom de la rubrique de retour

```
Exemple :
000 199 ZONE1
200 599 ZONE2
600 999 ZONE3
```

Les plages peuvent être définies jusqu'à 13 chiffres.

ATTENTION : Toutes les plages doivent avoir le même nombre de chiffres. Par exemple, si le code est défini sur 3 chiffres, une plage allant de 70 à 90 sera définie par 070 090...

M1CODERET M9CODERET [v1.03]

Idem MCODERET mais avec les mémos M1 à M9.

MCODETRANS [v1.00]

La fonction MCODETRANS renvoie un numéro de poste dans M1 en fonction d'un numéro contenu dans M2 lors de l'appel. Elle repose sur un fichier <nom_rubrique>.txt qui se trouve dans g4_dat\ . Son format est :

```
xxx yyy PPPP avec
```

xxx = début de la plage yyy = fin de la plage PPPP = numéro du poste cible

```
Exemple :
000 199 ZONE1
200 599 ZONE2
600 999 ZONE3
```

Lors du passage sur la rubrique appelante, M1 prendra pour valeur 2430 si la valeur dans M2 est comprise entre 000 et 199, etc... La taille du code d'accès doit être reprise dans le format du fichier .txt. La fonction MCODETRANS dérive de la fonction CODETRANS avec 2 différences:

- Le fichier contenant la table de transfert porte le nom de la rubrique appelante .TXT.

- Le numéro de sélection est contenu dans M2 et non dans le code d'accès. Une rubrique permettant l'entrée du code (code d'accès = ?????...) doit stocker ce dernier dans M2.

MCODEVERIF [v1.00]

La fonction MCODEVERIF utilise le fichier de contrôle portant le nom de la rubrique appelante suivi de l'extension .TXT (répertoire g4_dat). Cette fonction vérifie que le code présent dans M4 est défini dans le fichier de contrôle, si ce n'est pas le cas, elle exécute un renvoi sur la rubrique CODEERR. Si le code est présent, le retour se fait sans influence (l'arborescence continue normalement). Le format du code présent dans M4 n'a pas d'importance pour la librairie.

PRINTCOM [v1.00]

La fonction PRINT_COM permet de lancer la commande c:\SPXPRINT (un batch ou exe) en lui passant en paramètre la chaîne à sortir sur l'imprimante. SPXPRINT peut être un fichier .BAT possédant la commande suivante :

```
echo %1 %2 %3 %4 %5 %6 %7 %8 %9 >PRN
```

si l'imprimante est sur le port série COM2, on utilisera :

```
echo %1 %2 %3 %4 %5 %6 %7 %8 %9 >COM2 :
```

Il doit être précisé à Windows NT que SPXPRINT doit s'exécuter en Fenêtre Réduite avec l'option de Fermer en Quittant activée (voir Propriétés du fichier SPXPRINT.BAT) .

L'impression réalisée permet d'imprimer en temps réel les appels identifiés avec la date, l'heure et le numéro de téléphone du correspondant (tel qu'il s'est identifié sur des systèmes analogiques).

D'autre part, dans le cas d'une imprimante série, il est nécessaire de définir les paramètres de communication série dans le fichier AUTOEXEC.BAT à l'aide de la commande MODE. Par exemple, pour une imprimante OKI Microline 320 nécessitant un échange à 9600 Bauds sur 8 bits sans parité on utilisera :

```
MODE COM2 :96,n,8,1
syntaxe : port, vitesse de transmission, parité (Even/Odd/None), longueur, stop.
```

RETOURA [v1.11]

Cette fonction lit séquentiellement un nom de rubrique dans un fichier et effectue un retour dessus. Ce fichier est nommé : nom_de_rubrique.ret et se trouve dans g4_dat

Exemple de fichier:

```
-----
RUB_1
RUB_2
RUB_3
-----
```

Le premier appel à RETOURA, c'est RUB_1 qui est lu. Au second appel se sera RUB_2 etc...

RUBJOURMOIS [v1.16]

La fonction RUBJOURMOIS renvoie le nom de la rubrique correspondante à la date du jour. La liste des correspondances date-rubrique se trouve dans un fichier texte au format suivant : (répertoire g4_dat ou selon définition de configuration)

```
2507 RUBRIQUE
```

pour le 25 juillet par exemple

Note: Cette fonction était disponible en v1.13 à v1.15 sous le nom RUBRIQUEJOUR. En v1.16 et+ il faut utiliser le nouveau nom.

RUBJOUR [v1.16]

Idem RUBJOURMOIS mais la date n'est définie que par le jour.

SETLANGUE [v1.14]

La fonction SETLANGUE permet de sélectionner la langue (selon format ISO639-2/T) pour les fonctions espelle de G4.

Les 3 derniers caractères du nom de la rubrique appelante doivent correspondre au code langue selon la norme ISO639-2/T)

Note : Une version 2.11 ou ultérieure de G4 est nécessaire pour utiliser cette fonctionnalité.

TEL2FAX [v1.00]

Note : L'utilisation de cette fonction nécessite l'extension Serveur Télécopie Ouvvert sur PROVOX/AS-VOX.

La fonction TEL2FAX permet l'envoi d'un "ticket" sur un télécopieur distant indiquant le contenu du mémo M5. Cette fonction est principalement destinée à émettre un "ticket de contact" permettant à un correspondant de laisser ses coordonnées dans le but par exemple d'être rappelé.

Le "ticket" est de la forme :

```
-----DEBUT DU MESSAGE-----
```

```
CONTACT : 0243500600
```

```
-----FIN DU MESSAGE-----
```

La donnée transmise (ici le numéro de téléphone) doit être présentée dans le mémo M5.

Le numéro du télécopieur destination doit être paramétré dans PROVOX.INI

```
[signalisation]
faxdistant=xxxxxxx
```

Note : Selon le paramétrage du système il peut être nécessaire de préciser le préfixe de sortie (par exemple 0,) dans le paramétrage du numéro du télécopieur distant.

TEL2PRN [v1.00]

La fonction TEL2PRN permet l'envoi d'une page "ticket" sur l'imprimante par défaut du PC serveur. Cette fonction est principalement destinée à émettre un "ticket de contact" permettant à un correspondant de laisser ses coordonnées dans le but par exemple d'être rappelé. Le "ticket" est de la forme :

```
-----DEBUT DU MESSAGE-----
```

```
Appel sur serveur vocal le 01/03/2000 à 15:17:50:77
```

```
CONTACT : 0243500600
```

```
-----FIN DU MESSAGE-----
```

La donnée transmise (ici le numéro de téléphone) doit être présentée dans le mémo CallID.

Note:Le module TEL2PRN.EXE doit aussi être présent dans le sous-répertoire G4_BIN de AS-VOX G4.

TEL2PRNTLG [v1.00]

La fonction TEL2PRNTLG fonctionne comme TEL2PRN. Néanmoins, la donnée transmise doit être présentée dans le mémo M1. En outre, on dispose également d'une information supplémentaire qui est le nom de l'application appelant la fonction TEL2PRNTLG.

Note :Le module TEL2PRN.EXE doit aussi être présent dans le sous-répertoire G4_BIN de AS-VOX G4.

TESTM2NUL [v1.12]

Cette fonction teste la valeur de M2. Si M2 vaut zéro un retour est effectué sur la rubrique " VALNULLE", sinon c'est un retour sans influence.