

SOLAR

BIBIND

Bibliothèques d'interfaces industriels
pour langage FORTRAN

LOGICIEL

LOGICIEL

LOGICIEL

LOGICIEL

LOGICIEL



**BIBLIOTHEQUE DE SOUS-PROGRAMMES
D'INTERFACE FORTRAN
(PÉRIPHÉRIE INDUSTRIELLE)
MANUEL DE RÉFÉRENCE**

MARS 1982

1 164 383 00 036 02 / FR

SOMMAIRE	Pages
1 - INTRODUCTION	1 - 1
2 - CONTENU DE LA BIBLIOTHEQUE ET FONCTIONS DISPONIBLES	2 - 1
3 - ENTREES SORTIES ANALOGIQUES	3 - 1
3.1 - IMM 10 IMM 20	3 - 1
3.1.1 - Séquence d'appel	3 - 1
3.1.2 - Format du tableau des adresses physiques	3 - 2
3.1.3 - Modes d'échanges	3 - 2
3.1.4 - Syntaxe d'appel	3 - 3
3.2 - IML 05	3 - 3
3.2.1 - Séquence d'appel	3 - 3
3.2.2 - Format du tableau des adresses de voies	3 - 4
3.2.3 - Modes d'échanges	3 - 4
3.3 - SORTIES SUR AOL 04	3 - 4
3.3.1 - Séquence d'appel	3 - 5
3.3.2 - Format du tableau des adresses de voies	3 - 5
3.3.3 - Modes d'échanges	3 - 6
4 - ENTREES SORTIES NUMERIQUES	4 - 1
4.1 - ENTREES NUMERIQUES	4 - 1
4.1.1 - Séquence d'appel	4 - 1
4.1.2 - Format du tableau des adresses physiques (Numéros de voies)	4 - 2
4.1.3 - Modes d'échanges	4 - 2
4.2 - ENTRÉES NUMÉRIQUES SUR DIC 32	4 - 3
4.2.1 - Séquence d'appel	4 - 3
4.2.2 - Utilisation	4 - 3
4.2.3 - Format du tableau des adresses physiques	4 - 4
4.2.4 - Format du tableau buffer de réception	4 - 4

	Pages
4.3 - SORTIES NUMÉRIQUES	4 - 6
4.3.1 - Séquence d'appel	4 - 6
4.3.2 - Format du tableau des profils à sortir	4 - 7
4.3.3 - Modes d'échanges	4 - 7
4.3.4 - Syntaxe d'appel	4 - 8
4.4 - SORTIES NUMÉRIQUES MODULEES EN TEMPS	4 - 8
4.4.1 - Séquence d'appel	4 - 8
4.4.2 - Format du tableau des profils à sortir	4 - 9
4.4.3 - Modes d'échanges	4 - 9
4.4.4 - Syntaxe d'appel	4 - 10
5 - FONCTION ATTENTE DE FIN D'ÉCHANGE	5 - 1
5.1 - SÉQUENCE D'APPEL	5 - 1
5.2 - UTILISATION	5 - 1
5.3 - SYNTAXE D'APPEL	5 - 1
6 - REGLES D'UTILISATION	6 - 1
6.1 - BIBLIOTHEQUE BIBIND - ST	6 - 1
6.2 - NOMBRE DE PARAMETRES	6 - 1
6.3 - KSTORE	6 - 2
7 - EXEMPLE DE PROGRAMMATION	7 - 1
8 - EXEMPLE D'ÉDITION DE LIENS	8 - 1

1 - INTRODUCTION

Les sous-programmes décrits dans ce manuel, permettent un accès commode aux périphériques industriels (E/S analogiques, E/S logiques) à partir de programmes écrits en langage FORTRAN ou PL 16.

En ce qui concerne les programmes "BASIC", les sous-programmes sont utilisables à condition de respecter le format des paramètres d'appel (entier, tableau d'entier).

Les périphériques concernés :

Acquisitions analogique	:	chaîne de mesures IMLO5 chaîne de mesures IMM10 chaîne de mesures IMH20
Sortie analogique	:	module AOLO4
Acquisition T.O.R.	:	module DIL48 - DIO 37 - DIC 32 - Eii
Sortie T.O.R.	:	module DOL32 - DIO 37 - Eii

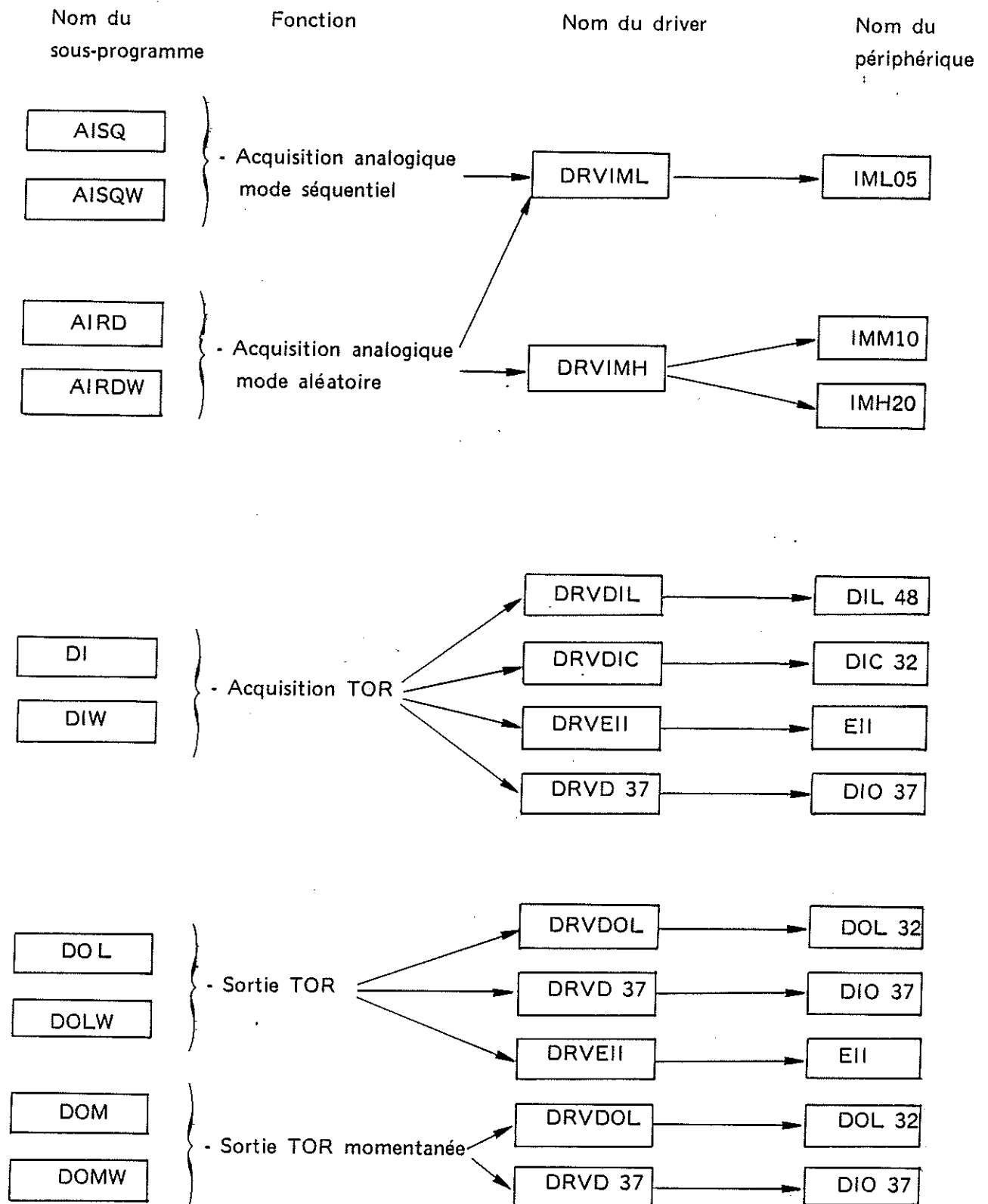
Mode d'échange :

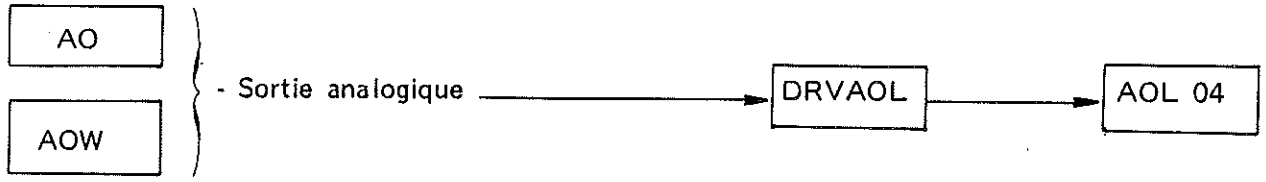
D'une façon générale, les interfaces dont le mnémonique se termine en "W" lancent un échange avec attente de la fin de l'opération demandée, sauf spécification contraire explicitement décrite.

2 - CONTENU DE LA BIBLIOTHEQUE BIBIND-ST

Nom du sous-programme	Fonction	Module Périphérique
AIRD	Acquisition analogique aléatoire retour immédiat	IML 05, IMM 10, IMH 20
AIRDW	idem : retour séquentiel	IML 05, IMM 10, IMH 20
AISQ	Acquisition analogique séquentielle retour immédiat	IML 05
AISQW	idem : retour séquentiel	IML 05
DI	Acquisition logique retour immédiat	DIL 48, DIO 37, DIC 32, EII
DIW	idem : retour séquentiel	DIL 48, DIO 37, DIC 32, EII
DOL	Sortie logique retour immédiat	DOL 32, DIO 37, EII
DOLW	idem : retour séquentiel	DOL 32, DIO 37, EII
DOM	Sortie logique momentanée retour immédiat	DOL 32, DIO 37
DOMW	idem : retour séquentiel	DOL 32, DIO 37
AO	Sortie analogique retour immédiat	AOL 04
AOW	Sortie analogique retour séquentiel	AOL 04
WEIO	Attente de fin d'E/S (doit suivre les appels AIRD, AISQ, AO, DOL).	

Les fonctions disponibles





WEIO Attente de fin d'échange

3 - ENTREES SORTIES ANALOGIQUES

3.1 - ENTREES ANALOGIQUES SUR IMM 10 OU IMM 20

AIRD : acquisitions analogiques aléatoires.
AIRDW : acquisitions analogiques aléatoires avec attente.

3.1.1 - Séquence d'appel

CALL $\left\{ \begin{array}{l} \text{AIRD} \\ \text{AIRDW} \end{array} \right\}$ (INV, JADR, KTAB, IERR)

INV = variable entière représentant le nombre de voies à scruter : n

JADR = tableau entier de dimension (n + 6) les six premiers mots sont réservés au système, la suite du tableau représente la commande n X (gain + n^o de voie)

KTAB = tableau entier de dimension n contenant les données après scrutation.

IERR = constante entière représentant le compte rendu de l'échange

IERR = 1 requête acceptée
= 2 erreur dans les paramètres de l'IOCB
= 3 défaut physique

3.1.2 - Format du tableau des adresses physiques

	Mode aléatoire		Mode séquentiel
JADR (0) ⋮	octet droit N° d'unité d'échange		X
JADR (5)	Réservé au système		
JADR (6) ⋮	Gain	N° de la 1ère voie	
⋮		⋮	
JADR (n)	Gain	N° de la nième voie	

- le premier mot du tableau des numéros de voies contient dans l'octet droit le numéro de l'unité d'échange FU ou SU.
- le bloc de contrôle IOCB occupe les mots 0 à 5 du tableau des commandes JADR.
- l'appelant précisera le N° d'unité d'échange utilisée

3.1.3 - Modes d'échanges

Dans le cas de AIRDW, le retour se fait toujours en fin d'échange, dans le cas de AIRD, le retour se fait après initialisation de l'échange.

Le driver gère le module en mode canal.

Le mode d'échange adopté est le mode aléatoire.

3.1.4 - Syntaxe d'appel

- En PL 16 :

```
CALL AIRD (INV, JADR, KTAB, IERR, RA := 4)
```

Le registre RA doit en effet, contenir le nombre de paramètres qui sont transmis lors de l'appel.

- En fortran :

```
CALL AIRD (INV, JADR, KTAB, IERR)
```

3.2 - ENTREES ANALOGIQUES SUR IML 05

AISQ : acquisition analogiques séquentielles.

AISQW : acquisitions analogiques séquentielles avec attente.

3.2.1 - Séquence d'appel

```
CALL { AISQ } (INV, JADR, KTAB, IERR)
     { AISQW }
```

INV = une variable entière représentant le nombre de voies à scruter : n

JADR = tableau entier de dimension 7. Les mots 0 à 5 sont réservés au système, le mot 6 représente le numéro de la première voie à scruter.

KTAB = un tableau entier de dimension n donnant les données après la scrutation

IERR = une variable entière représentant le compte rendu de l'échange

```
IERR = 1  requête acceptée
      = 2  erreur dans les paramètres de l'IOCB
      = 3  défaut physique.
```

3.2.2 - Format du tableau des adresses des voies

	Mode aléatoire	Mode séquentiel
JADR (0)	Octet droit = n° de l'unité d'échange	X
⋮	Réserve au système	
JADR (5)		
JADR (6)	N° de la 1ère voie	
⋮		
JADR (n)	N° de la nième voie	

L'octet droit du mot 0 contient le numéro de l'unité d'échange, le bloc de contrôle IOCB occupe les mots 0 à 5 du tableau de commande JADR.

3.2.3 - Modes d'échanges

Pour AISQW, le retour se fait en fin d'échange, pour AISQ, il se fait après l'initialisation de l'échange.

Le driver gère le module en programmé prioritaire.

Le mode d'échange adopté est le mode séquentiel.

La syntaxe d'appel est la même que celle décrite en 3.1.4.

3.3 - SORTIES SUR AOL 04

AO : Sortie analogique sans attente.

AOW : Sortie analogique aléatoire avec attente.

3.3.1 - Séquence d'appel

CALL $\left\{ \begin{array}{l} \text{AO} \\ \text{AOW} \end{array} \right\}$ (INV, JADR, KTAB, IERR)

INV = variable entière représentant le nombre de voies à scruter : n

JADR = tableau entier de dimension n + 6, les mots 0 à 5 sont réservés au système, la suite du tableau représente la commande (N° de voie)

KTAB = tableau entier de dimension n l'utilisateur fournit des valeurs analogiques dans cette table

IERR = variable entière représentant le compte rendu de requête

IERR = 1 requête acceptée
 = 2 erreur dans le paramètre de l'IOCB
 = 3 défaut physique.

3.3.2 - Format du tableau des adresses de voies

	Mode aléatoire	Mode séquentiel	
JADR (0)	X	Octet droit : N° d'unité d'échange	
		Réservé au Système	
JADR (5)			
JADR (6)		Gain	N° de la 1ère voie

- Le premier mot du tableau des numéros de voies contient dans l'octet droit le numéro de l'unité d'échange.

- Le bloc de contrôle d'IOCB occupe le mot 0 à 5 du tableau de commande JADR.
- L'appelant précisera le numéro d'unité d'échange utilisé.

3.3.3 - Modes d'échanges

Le driver gère le module en mode programmé prioritaire:

Le mode d'échanges adopté est le mode séquentiel.

La syntaxe d'appel est identique à celle décrite en 3.1.4.

4 — ENTREES SORTIES NUMERIQUES

4.1 - ENTREES NUMERIQUES

DIL 48 - DIO 37 - EII

DI : acquisition numérique

DIW : acquisition numérique avec attente.

4.1.1 - Séquence d'appel

CALL $\left\{ \begin{array}{l} DI \\ DIW \end{array} \right\}$ (INV, JADR, KTAB, IERR)

INV = variable entière représentant le nombre de voies à scruter (n).

JADR = tableau entier de dimension (n + 6). Les 6 premiers mots sont réservés au système, la suite du tableau représente les numéros des voies à scruter (uniquement pour les modules DIL 48).

KTAB = tableau entier de dimension M contenant les données après scrutation

IERR = constante entière représentant le compte rendu de l'échange :

IERR = 1 requête acceptée

= 2 erreur dans les paramètres de l'IOCB

= 3 défaut physique.

4.1.2 - Format du tableau des adresses physiques (Numéros de voies)

	Mode aléatoire	
JADR (0)	Octet droit : N° d'unité d'échange	DIO 37/EII
JADR (5)	Réservé au système	
JADR (6)	N° de la 1ère voie	DIL 48
JADR (n)	N° de la nième voie	

- L'octet droit du premier mot contient le numéro de l'unité d'échange.

4.1.3 - Modes d'échanges

Dans le cas de DI aussi bien que DIW, les acquisitions sont faites avec retour en fin d'échange étant donné que le driver gère le module en programmé simple.

Le mode d'échange adopté est le mode aléatoire:

La syntaxe d'appel est la même que celle décrite en 3.1.4.

Remarques :

- a) Utilisation avec le module DIO 37 : le sous-programme ne permet la lecture que des deux mots (le paramètre INV a impérativement pour valeur 3).
- b) Utilisation avec le module EII : le paramètre INV est égal à 4 fois le nombre de mots à lire.

Δ

4.2 - ENTRÉES NUMÉRIQUES SUR DIC 32

DI : acquisition numérique
DIW : acquisition numérique avec attente

4.2.1 - Séquence d'appel

CALL DI (INV, JADR, KTAB, IERR)
DIW

INV variable entière représentant le nombre de mots à scruter (est égal à 2 fois le nombre de coupleurs n)

JADR tableau entier de dimension (n + 6). Les 6 premiers mots sont réservés au système. La suite du tableau représente :

- en mode programmé simple, la table des modules,
- en mode programmé prioritaire, la table d'initialisation.

KTAB tableau entier de dimension m correspondant au buffer de réception.

IERR constante entière représentant le compte rendu de l'échange :

IERR = 1 requête acceptée
= 2 erreur dans les paramètres de l'IOCB
= 3 défaut physique

4.2.2 - Utilisation

Le module permet de faire des acquisitions suivant les deux modes de fonctionnement :

- en mode programmé simple,
- en mode programmé prioritaire.

Pour plus de détails, on se reportera au manuel d'exploitation DIC 32 (1 159 601 00 046 04).

4.2.3 - Format du tableau des adresses physiques

	Programmé simple		Programmé prioritaire	
JADR (0)	Octet fonction = '00	N° unité d'échange	Octet fonction = '20	N° unité d'échange
JADR (1)	Réservé système		Réservé système	
JADR (5)	Réservé système		Réservé système	
JADR (6)	N° du 1er coupleur à scruter		Masque IT du 1er coupleur	
JADR (n)	N° du n ^{ième} coupleur à scruter		Masque IT du n ^{ième} coupleur	

L'octet gauche du premier mot contient le mode de fonctionnement :

- en mode programmé simple '00
- en mode programmé prioritaire '20.

L'octet droit du premier mot contient le numéro de l'unité d'échange.

4.2.4 - Format du tableau buffer de réception

	Programmé simple	Programmé prioritaire
KTAB (0)	X	Pointeur de remplissage
KTAB (1)		Pointeur de vidage
KTAB (2)		m = longueur de la queue circulaire de réception
KTAB (3)		
KTAB (4)		
KTAB (5)		
KTAB (m)		
KTAB (m + 1)		
KTAB (m + 2)		

Δ

) En mode programmé simple, la longueur du tableau est égale à 2 fois le nombre de coupleurs à scruter (= INV).

En mode programmé prioritaire, la longueur du tableau est égale à $m + 3$ (m multiple de 3) et dimensionnée par l'appelant.

L'appelant doit initialiser le pointeur de remplissage, le pointeur de vidage et la longueur de la queue circulaire.

La gestion du pointeur de vidage est à la charge de l'appelant. La tâche en attente de changements d'états doit se terminer par :

CALL EXIT

) Lorsque le pointeur de remplissage rattrape le pointeur de vidage, le 4ème mot de JADR est positionnée à 1 (JADR (3) = '0001).

4.3 - SORTIES NUMÉRIQUES

DOL 32 - DIO 37 - EII

DOL : Sorties numériques sélectives sans attente

DOLW : Sorties numériques sélectives avec attente

4.3.1 - Séquence d'appel

CALL $\left\{ \begin{array}{l} \text{DOL} \\ \text{DOLW} \end{array} \right\}$ (INV, JADR, KTAB, MTAB, IERR)

INV = variable entière représentant le nombre d'octets à sortir (est égal à 2 fois le nombre de mots)

JADR = Tableau entier de dimension $n + 7$. Les 7 premiers mots sont réservés au système, la suite du tableau représente les numéros des mots à sortir (uniquement pour les modules DOL 32).

KTAB = Tableau entier de dimension 4 contenant le profil des mots à sortir.

MTAB = Tableau entier de dimension 4 contenant les bits sur lesquels on veut agir (le driver sort l'intersection de KTAB et de MTAB).

IERR = constante entière représentant le compte rendu de l'échange :

IERR = 1 requête acceptée
= 2 erreur dans les paramètres
= 3 défaut physique.

4.3.2 - Format du tableau des profils à sortir

MODE ALEATOIRE		DIO 37/EII	DOL 32
JADR (0)	Octet droit = N° Unité d'échange		
JADR (1)	INDIFERRENT = MODULE EII 0 = MODULE DIO 37 1 = MODULE DOL 32		
JADR (2)	Réservé au système		
⋮			
JADR (6)	Réservé au système		
JADR (7)	Profil du 1er mot		
⋮			
JADR (n)	Profil du n ème mot.		

L'octet droit du premier mot contient le numéro d'unité d'échange.

4.3.3 - Modes d'échanges

Les sorties DOL et DOLW se font toujours avec retour en fin d'échange : les modules DOL 32 et DIO 37 fonctionnant en programmé simple.

Dans le cas des modules DOL 32 le mode d'échange adopté est le mode aléatoire.

Remarques :

Utilisation avec le module EII : le paramètre INV est égal à 4 fois le nombre de mots à sortir. Bien que le driver ne permette pas d'effectuer des sorties sélectives, la présence du paramètre MTAB est obligatoire.

4.3.4 - Syntaxe d'appel

- En PL 16 :

```
CALL DOL (INV, JADR, KTAB, MTAB, IERR, RA := 5) ;
```

- En fortran :

```
CALL DOLW (INV, JADR, KTAB, MTAB, IERR).
```

4.4 - SORTIES NUMERIQUES MODULEES EN TEMPS

DOL 32 - DIO 37

DOM : Sorties numériques sans attente modulées en temps

DOMW : Sorties numériques avec attente modulées en temps.

4.4.1 - Séquence d'appel

```
CALL { DOM } (INV, JADR, KTAB, MTIME, IERR)  
     { DOMW }
```

INV = variable entière représentant le nombre d'octets à sortir (est égal à 2 fois le nombre de mots).

JADR = Tableau entier de dimension $n + 7$. Les 7 premiers mots sont réservés au système, la suite du tableau représente les numéros des mots à sortir (uniquement pour les modules DOL 32).

KTAB = Tableau entier de dimension n contenant le profil des mots à sortir.

MTIME = Durée de sortie en nombre de tops de base de l'horloge.

IERR = constante entière représentant le compte rendu de l'échange :

IERR = 1 requête acceptée
 = 2 erreur dans les paramètres de l'IOCB
 = 3 défaut physique.

4.4.2 - Format du tableau des profils à sortir

MODE ALEATOIRE		DIO 37/EII	DIO 32
JADR (0)	Octet droit : N° Unité d'échange		
JADR (1)	Réservé au système		
⋮			
JADR (6)	Réservé au système		
JADR (7)	Profil du 1er mot		
⋮			
JADR (n)	Profil du nième mot		

L'octet droit du premier mot contient le numéro d'unité d'échange.

4.4.3 - Modes d'échanges

Les sorties DOM et DOMW se font toujours avec retour en fin d'échange : les modules DOL 32 et DIO 37 fonctionnent en programmé simple.

Dans le cas des modules DOL 32, le mode d'échange adopté est le mode aléatoire.

Le principe des sorties modulées est le suivant :

- * Les sorties sont maintenues aux profils indiqués dans le tableau JADR pendant le temps MTIME.
- * Lorsque ce délai est écoulé toutes les sorties sont positionnées à 0.

Remarques :

Les sous-programmes DOM et DOMW ne permettent pas une utilisation avec les modules EII. Il est quand même possible d'effectuer des sorties momentanées au moyen de la séquen-

ce suivante :

```
CALL DOL (INV, JADR, KTAB, MTAB, IERR) < Sortie des profils
CALL WAIT (ID, IVD, IERR) < temporisation
CALL DOL (INV, JADR, KTAB, MTAB, IERR) < remise à zéro du profil.
```

4.4.4 - Syntaxe d'appel

- En PL 16 :

```
CALL DOMW (INV, JADR, KTAB, MTIME, IERR, RA := 5) ;
```

- En fortran :

```
CALL DOM (INV, JADR, KTAB, MTIME, IERR).
```


5 - FONCTION ATTENTE DE FIN D'ÉCHANGE

WEIO Attente de fin d'échange.

5.1 - SEQUENCE D'APPEL

CALL WEIO (JADR, IERR)

JADR : Tableau entier utilisé pour l'échange dont on veut attendre la fin.

IERR. : Constante entière représentant le compte rendu :

IERR = 1	Echange terminé normalement
= 2	Erreur de paramètre
> 2	Défaut physique:

5.2. - UTILISATION

Ce sous-programme est à utiliser après les appels à AIRD, AISQ, AO et DOL. Il permet d'attendre la fin d'échange et être informé des conditions dans lesquelles il s'est déroulé.

Pour AIRD et AISQ, il doit obligatoirement précéder le traitement des mesures acquises.

Le paramètre JADR doit être le même que celui utilisé par l'appel de AIRD, AISQ AO ou DOL.

5.3 - SYNTAXE D'APPEL

Ne pas oublier en PL 16 RA := 2.,

6 — REGLES D'UTILISATION

6.1 - BIBLIOTHEQUE BIBIND - ST

Les différents sous-programmes sont livrés sous forme d'une bibliothèque qui est à scruter lors de l'édition de liens de la tâche qui les utilise.

6.2 - NOMBRE DE PARAMETRES

Le sous-programme vérifie les paramètres spécifiés lors de l'appel ainsi que leur nombre. Il tient compte également du compte rendu d'IOCS concernant la validité de l'IOCB fabriqué à partir des paramètres de l'appel.

En cas d'erreur, le compte rendu est 2.

Mais si le nombre de paramètres est incorrect, le sous-programme d'interface n'est pas capable d'accéder à IERR.

Dans ce cas, il ne range aucune valeur dans les tables et les mémoires de l'appelant.

Il est donc conseillé à l'utilisateur d'initialiser IERR à -1 et de tester le compte rendu comme suit :

IERR = 1	requête acceptée.
= 2	erreur dans les paramètres (leur nombre est cependant correct).
= 3	défaut physique.
= -1	Nb paramètres incorrects.

6.3 - KSTORE

Les 4 paramètres sont empilés dans la KSTORE de l'appelant.

D'autre part, en comptant l'appel à IOCS et les sauvegardes de bases, il faut compter 40 mots de KSTORE pour ces différents sous-programmes.

7 — EXEMPLE DE PROGRAMMATION

- En PL 16 :

MAIN PROCEDURE TESTCH

. LOCAL SECTION LOC

RES 2 ;

REF PROCEDURE AIRD ;

WORD IERR ;

WORD INV = (5) ;

ARRAY 11 WORD JADR ;

ARRAY 5 WORD KTAB ;

⋮

. USING RL = LOC ;

⋮

Préparation de la table JADR (@ voies, gain).

⋮

IERR := -1 ;

CALL AIRD (@INV, @JADR, @KTAB, @IERR, RA := 4) ;

RA := IERR ;

GOTO PAOK ON (RA/=1) ;

IERR := -1 ;

CALL WEIO (@JADR, @IERR, RA := 2) ;

RA := IERR ;

GOTO PAOK ON (RA/= 1) ;

⋮

. En Fortran :

```
SUBROUTINE TESTCH  
COMMON IERR  
COMMON INV  
COMMON JADR (11)  
COMMON KTAB (5)
```

⋮

```
10 CONTINUE
```

⋮

Préparation de la table JADR (α)voies, gain).

⋮

```
INV = 5  
IERR = -1  
CALL AIRD (INV, JADR, KTAB, IERR)  
IF (IERR. NE. 1) CALL ERREUR  
IERR = -1  
CALL WAIO (JADR, IERR)  
IF (IERR. NE. 1) CALL ERREUR
```

⋮

8 - EXEMPLE D'EDITION DE LIENS

```
/ JOB      XXXX ,, D3
/ CALL    EDILE
/ BI      TESTCH - LK
/ BO      TESTCH - MO
/ OPEN    OLD 3, BIBIND - ST
/ ILNK
  :
/ LLNK 3
  :
/ link éventuel d'autres bibliothèques ou S/P
  :
/ RLNK
/ ELNK
/ CALL BUILD
/ SLOD
  :
/ CATAL IM, TESTCH - IM
/ EOJ
```

Distribution codes/Codes de diffusion			
Customers : Clients :			
Internal : Interne :			

DELIVERY ADDRESS
ÉTIQUETTE ADRESSE

Bull MTS

1, Rue de Provence

B.P. 208

38432 ÉCHIROLLES CEDEX / FRANCE

